U.S. ARMY MISSILE RESEARCH AND DEVELOPMENT COMMAND

Redstone Arsenal, Alabama 35809

TECHNICAL REPORT T-79-43

TRI-FAST HARDWARE-IN-THE-LOOP SIMULATION

Volume I: Tri FAST Hardware-in-the-Loop Simulation at the Advanced Simulation Center

David B. Merriman
Robert C. Ward
Systems Simulation Directorate
Technology Laboratory
Redstone Arsneal 35809

James T. Taylo
Army Systems Division
Boeing Aerospace Company
Huntsville, Alabama 35809

and

C. R. Green
Computer Sciences Corporation
Huntsville, Alabama 35809

28 March 1979

Approved for Public Release; Distribution Unlimited.

DMI FORM 1000, 1 APR 77

DISPOSITION INSTRUCTIONS

DESTROY THIS REPORT WHEN IT IS NO LONGER NEEDED. DO NOT RETURN IT TO THE ORIGINATOR.

DISCLAIMER

THE FINDINGS IN THIS REPORT ARE NOT TO BE CONSTRUED AS AN OFFICIAL DEPARTMENT OF THE ARMY POSITION UNLESS SO DESIGNATED BY OTHER AUTHORIZED DOCUMENTS.

TRADE NAMES

USE OF TRADE NAMES OR MANUFACTURERS IN THIS REPORT DOES NOT CONSTITUTE AN OFFICIAL ENDORSEMENT OR APPROVAL OF THE USE OF SUCH COMMERCIAL HARDWARE OR SOFTWARE.

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER T-79-43	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) 6 TRI-FAST HARDWARE-IN-THE-LOOP SIMULATION. Volume I. Tri-FAST Hardware-in-the-Loop Simulation at the Advanced Simulation Center.		5. TYPE OF REPORT & PERIOD COVERED 9 Technical Report. 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) 10 David B. Merriman, Robert C. Ward James T. Taylor, C. R. Green		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Commander US Army Missile Research and Development Command ATTN: DRDMI-TDD Redstone Arsenal, Alabama 35809		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Commander US Army Missile Research and Development Command ATTN: DRDMI-TI Redstone Arsenal, Alabama 35809		12. REPORT DATE 11 28 March 1979 13. NUMBER OF PAGES 82
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) 12 85p.		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release; Distribution Unlimited.

14 DRDMI-T-79-43-VOL-1

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Prepared in cooperation with:

Computer Sciences Corporation, Huntsville, Alabama 35809 and Army Systems Division, Boeing Aerospace Company, Huntsville, Alabama 35809

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Tri-FAST	Hardware-in-the-Loop
ACSL	Advanced Simulation Center
Simulation	RF Target Models

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The purpose of this report is to document the Tri-FAST missile simulation development and the seeker hardware-in-the-loop (HWIL) testing in the Radio Frequency Simulation Cell of the Advanced Simulation Center (ASC). Tri-FAST is a tri-service active seeker technology program.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

393 427

A

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

CONTENTS

CONTENTS (Concluded)

ILLUSTRATIONS

ILLUSTRATIONS (Concluded)

1. INTRODUCTION

The purpose of this report is to document the Tri-FAST missile simulation development and the seeker hardware-in-the-loop (HWIL) testing in the Radio Frequency Simulation Cell of the Advanced Simulation Center (ASC). Tri-FAST is a tri-service active seeker technology program. Simulations included a Baseline Digital used for missile math model implementation verification, a Hybrid Only simulation for analog dynamic check-out, a Real Time Digital HWIL simulation for checking out the missile simulation/Radio Frequency Simulation System (RFSS) interface and a Real Time Hybrid HWIL simulation for closed-loop production testing with the Tri-FAST hardware seeker actually tracking RF targets and providing guidance commands to the missile simulation. Testing was done with air-to-air and surface-to-air scenarios provided by the Naval Weapons Center (NWC), China Lake, California. Math models for both the missile simulation and RF targets were also provided by NWC. The seeker math model was developed by Motorola, contractor for the seeker. The missile simulations used at ASC were developed by the Systems Dynamics Branch of the Systems Simulation Directorate. The ACSL simulation language was used for the digital portions of the missile simulations. The RF target generation hardware and software were developed in the RFSS.

2. GENERAL DISCUSSION

The Tri-FAST simulation documentation is broken into five volumes. A brief description of the topics discussed in each volume follows.

Volume I. Tri-FAST Hardware-in-the-Loop Simulation at the Advanced Simulation Center. The development task for the missile leading to a hardware-in-the-loop simulation is discussed as are the various simulation configurations. A narrative of the signal flow through the Hybrid HWIL is included. HWIL production tests are described, including the output data, RFSS coordinate systems, targets and scenarios run.

Volume II. Tri-FAST Mathematical Modeling Description. The math models for the missile simulation are elaborated in great detail; units and coordinates are defined.

Volume III. Tri-FAST Real Time Digital/RFSS, Real Time Hybrid/RFSS and Hybrid Only Simulations. The additional software added to the Baseline Digital simulation to run the Real Time Digital hardware-in-the-loop and Real Time Hybrid hardware-in-the-loop simulations is documented here. The Hybrid Only documentation is also in this volume. All of the appendices referenced from volumes with no appendices attached are in Volume III. The appendices include analog diagrams and listings of the real time digital programs for the Real Time Digital/RFSS, Real Time Hybrid/RFSS and Hybrid Only simulations.

Volume IV. Tri-FAST Simulation Verification. The verification tests accomplished for the Baseline Digital, Real Time Digital hardware-in-the-loop, Real Time Hybrid hardware-in-the-loop and Hybrid Only simulations are described.

Volume V. ACSL Postprocessor Plots for Tri-FAST Hardware-in-the-Loop Simulation Data. ACSL Postprocessor plots of the hardware-in-the-loop production test data on digital tape comprise Volume V.

3. SIMULATION DEVELOPMENT

A. INTRODUCTION

The ASC Tri-FAST simulation program required the development of missile simulation software for the CDC 6600 and missile simulation analog hardware to close the loop with the RFSS for hardware-in-the-loop (HWIL) tests. The RFSS contained the hardware seeker and seeker interface. The RFSS hardware/software drove the three-axis rotational table on which the seeker was mounted; provided the RF target and environment for the seeker; acted as the interface between the CDC 6600 and analog computers and the hardware seeker; and was the simulation executive controller. Typically, for time-critical HWIL operation a hybrid computer simulation is required. The majority of the missile math models are usually implemented on the analog computer, whereas the digital computer is used to control the simulation, record data, perform table look-ups for variables that have low update rates, and perhaps implement the missile translational equations. At ASC, however, the real time digital computer, a CDC 6600, has a large memory and operates at such high speeds that real time, time-critical HWIL missile simulations are usually possible. Depending on the digital integration step size requirements, it is possible to run HWIL tests with a real time, all digital missile simulation with either simplified or "high fidelity" missile models. Thus, two options are generally available for running HWIL tests at ASC. Each has its own advantages and both have been used in the past. For Tri-FAST, both options were implemented and utilized.

B. BASELINE DIGITAL

The simulation developmen lways starts with an all digital implementation of the ailed math models. With the ACSL simulation language m lar approach, missile models that are usually common to all missiles (for example, the target and atmosphere models) are available at ASC in ACSL code and can be inserted literally as a block into a new simulation program. This modular method of programming with ACSL allows implementation of a physical subsystem model (for example, an actuator or seeker) as one contiguous block of code without regard for the order in which the variables must be calculated to ensure that all elements on the right-hand side of an equation are defined. The end result is fast implementation with minimum debugging and a logical, easily read signal flow through the subsystem models.

After the simulation has been programmed and debugged, model verification by comparison with other "standard" missile simulations for this particular system is accomplished, or model validation is initiated using subsystem hardware test data and/or flight and chamber test data. For Tri-FAST, ASC accomplished model verification by comparison with other simulations. In this approach, open-loop tests are performed on the ACSL subsystem modules. Then open-loop comparison checks are made in a staircase fashion with an increasingly more complex set of modules. Finally, the missile simulation is verified by a full closed-loop missile scenario. It is in the verification task that the efficiency of the ACSL modular method of combining subsystem models becomes the most noticeable. Test programs can be applied to one or a combination of the subsystem modules to perform open-loop verification tests without reprogramming the modules. Typically, the assembly language program that performs table look-up functions for the aerodynamic models and the ACSL missile rotational module are checked out first. Actuator-only module tests are performed and the actuator and missile translational modules are integrated with the aerodynamics and rotational module for check-out. The autopilot module is then added to the above subsystems and the combined configuration is verified. Finally, seeker-only module tests are conducted, and the seeker and target modules are added to the simulation for closed-loop tests with chosen scenarios.

C. MULTI-DERIVATIVE DIGITAL

After verification, the modules are divided into two or three separate DERIVATIVE sections with different integration step sizes and perhaps different integration algorithms. This capability is peculiar to ACSL and originated from a request of ASC's Modeling and Analysis Group to the authors of ACSL. The models are separated based on their integration step size requirements for numerical stability and on their rates of change of values. For instance, the density of air varies slowly with respect to time and is usually updated every 50 msec, whereas a first order transfer function with a 1-msec time constant must be integrated with a step size no greater than 1 msec. The ACSL multi-derivative digital program is a prelude to either the Hybrid HWIL or the All Digital HWIL.

For the Hybrid HWIL, multi-derivative programming is used to design the partitioning of models between the digital and analog machines. The integration step size selected for the proposed analog modules is made small enough in comparison to the step size of the proposed digital modules to make the analog section variables appear continuous to the digital section. Compensation is necessary to correct for the resulting time skewing between the sections. Such skewing is quite similar to that in the actual hybrid. An indication of the compensation requirements for the hybrid is a major result of the multi-derivative program. For the All Digital HWIL, multi-derivative programming makes more efficient use of execution time by less frequent updating of variables that vary slowly with time. Compensation is also needed in the all digital. For the Hybrid HWIL, the hardware seeker signals are sent continuously to the analog autopilot so that compensation is not required at that interface. However, for the All Digital HWIL the effects of a discrete update rate from the seeker must be assessed. For Tri-FAST, only a 10-msec DERIVATIVE section was used for the math models for both the All Digital and Hybrid HWIL simulations. Subsystem software modules are not usually broken apart and put in different DERIVATIVE sections since the modules were defined with a view to multi-derivative and hybrid programs. The major consideration is the choice of and compensation required by the interface variables (both analog/digital and DERIVATIVE section/DERIVATIVE section interfaces).

D. REAL TIME DIGITAL

Three ACSL real time digital programs are required. For brevity, these programs will be referred to as follows in the discussion below:

- A digital - the ACSL All Digital HWIL simulation (the Real Time Digital HWIL)

- B digital - the ACSL digital portion of the Hybrid HWIL simulation

- C digital - the ACSL digital portion of the Hybrid Only simulation.

The All Digital HWIL, A digital, and the digital for Hybrid HWIL, B digital, were discussed above. For Tri-FAST, the Hybrid Only real time digital program, C digital, contained a simplified seeker to provide closed-loop dynamic analog check-out and follow-on analog maintenance. The analog simulation was then interfaced with the B digital to form the Hybrid HWIL that was to be the main simulation during production testing of the hardware seeker. The purpose of the A digital for Tri-FAST was to be interface check-out with the RFSS (both with and without the hardware seeker).

In developing the B digital, a special real time ACSL library was used which interfaces ACSL with the special CDC 6600 real time FORTRAN language additions. The B digital had to communicate with the seeker hardware, the ASC hardware used to implement a realistic target and environment for the seeker hardware sensors, and the three-axis rotation table on which the seeker hardware was mounted to simulate missile body rotations in flight. The three-axis flight table and the RF target and RF environment for the seeker hardware reside in the RFSS. The RFSS master computer (the Datacraft) controls the RF target and assc iated environment through minicomputers and drives the three-axis table. The RFSS master computer communicated to the CDC 6600 and consequently the B digital through a special interface called Direct Cell C which makes a portion of the RFSS master computer memory look like additional storage to the 6600. Consequently, the 6600 is capable of reading and writing data to the RFSS master computer just as it would to its own memory. The B digital had to communicate with one or more analog machines through very high speed analog-to-digital and digital-to-analog converters (ADC's and DAC's). To ensure that information was transferred between the RFSS and the B digital and between the B digital and analog at the desired discrete time intervals, an interrupt system in the CDC 6600 made available one or more interrupts for sending serial input/output data across the Direct Cell C or the ADC's and DAC's. Thus, the B digital had to have interrupt code inserted, special CDC memory locations (actually the RFSS computer memory) declared for Direct Cell communications, and scaling and decoding equations for both Direct Cell and ADC/DAC I/O variables. The B digital has two basic parts, the real time segment(s) and the batch background. In general there is a real time segment for each ACSL Derivative section and correspondingly one interrupt for each segment. The real time segment(s) incorporates the above and the missile modeling. Once it starts executing in real time, no other job interferes with its running. At the end of a run the batch background portion of the ACSL program copies data that were generated by the real time segment to disk files. The time interval left after the real time segment has finished execution for an interrupt period and before the next interrupt will occur is available for processing batch jobs. It is during this time period that the batch background is executed. However, it competes with other batch jobs for execution time.

The A digital and C digital had some of the same programming considerations as described above. But the A digital interfaced only with the Direct Cell, whereas the C digital interfaced only with the analog computers. Both, however, had to be designed with interrupts and a batch background program. As for transforming the Baseline Digital into a real time program for communicating with hardware, one can see that the A digital and C digital program design considerations are subsets of those for the B digital. The split in the math models between hardware and software is obviously different for the three real time designs. Block diagrams and accompanying discussions on this subject are deferred to Subsection 3.I. Details of the real time digital programs can be found in Volume III.

E. SIMULATION DATA RETRIEVAL

For the Hybrid HWIL simulation production runs, data recording was done in the real time digital (see Volume III, Subsection 2.C). The digital real time segment(s) records data during real time into what is called the CDC 6600 Extended Core Storage (ECS). The digital batch background copies the data from ECS to disk. After the simulation run, ACSL can provide a quick look presentation of the raw data using its standard run time commands for displaying, printing, and plotting. The raw data were copied from disk to magnetic tape, and a copy was provided to NWC and Motorola.

An ACSL Postprocessor program was implemented to perform analysis of Monte Carlo data and to transform data to units and coordinates desired by the customer.

F. ECSSL PROGRAM AND HYBRID PARTITIONING

There are numerous ways to partition a simulation into analog and digital portions and achieve satisfactory simulation performance. The architecture should allow for expansion capability in all areas, allow use by other real time tasks, and be highly automated for setup and check-out. As a result of math model analyses (such as results from the ACSL multi-derivative program) the portion of the simulation to be made analog is decided upon and this portion is further partitioned among the various analog hardware components. For Tri-FAST, the models were split between the EAI 781 and AD-4 computers with table look-ups performed by the hybrid Multi-Variable Function Generator (MVFG), which is an entity in itself. The actual split in modeling is discussed in Subsection 3.I. EAI and ASC have implemented and will be implementing software that relieves the analog program designer from much of the programming manual labor and will also do most of the straightforward analog design. The major software package of this sort is the ECSSL compiler which is run at ASC on the CDC 6600. The user inputs to the ECSSL compiler the math models (in a specific form); the MVFG or analog computers on which the models are to be implemented; variable types; scale factors; the simulation constraints; data tables; assignments of trunks for communication among the analog computers, CDC 6600 and RFSS; the static check values, etc. ECSSL transforms these data into their equivalent analog hardware representation. It generates the scaled analog equations, assigns specific analog components (for example, integrators, function generators) for these equations, shows how these components are connected, calculates static check values and generates breakpoint data for look-up tables. The compiler also generates a HYTRAN program that the ASFISS (Advanced Simulation Facility Interconnection and Setup Subsystem) controller uses to automatically set DCA's and servo pots for automated on-line (static check) verification as well as providing for off-line verification of all patchboard interconnections and most of the trunk line connections. The HYTRAN program through ASFISS provides automatic setup on the MVFG. The HYTRAN program listing is in a suitable form to serve as instructions for board patching. Another software feature available at ASC is the Analog Diagrammer which uses the HYTRAN program stored on disk as the information source for automated generation of analog interconnection diagrams and component cross reference lists. Trunking Station Utility software generates trunking station patch lists to ensure proper connection of major hardware items in an atmosphere where there may be two or three different hybrid simulations sharing common equipment such as the CDC 6600.

G. ANALOG IMPLEMENTATION

In its present configuration ECSSL does not include in its design any logic programming, mode control or continuous resolver operation, but in the near future all of these will be available in ECSSL. For now, these operations are designed and implemented by the analog programmer. Additional modifications are made to the implemented simulations and comparable changes are made to the HYTRAN program to reflect improvements in gain distribution, optimization of scaling and explicit representation of variables not previously defined in a manner to allow display.

After the AD-4 and EAI 781 boards are patched, the ASFISS controller automatically sets the DCA's and servo-set potentiometers and performs a static check on the two computers to assure correct interconnections as well as assuring satisfactory operation of the passive components, summers, multipliers, etc. No dynamic check of the operation of integrators is made with ASFISS. Any out-of-range checks are printed in an error summary list to allow correction of the errors.

Dynamic checks are developed for each analog model module. Test results are compared with overlay plots generated with the verified baseline digital modules. Other checks like frequency response tests of filters are made. Then, various combinations of the modules are tested together and again checked against the verified digital.

H. HYBRID INTEGRATION AND CHECK-OUT

During hybrid integration and check-out, the CDC 6600 real time digital program for Hybrid Only (C digital) simulation with the math models (Subsection 3.I) and simulation control necessary for closed-loop operation are mated with the analog hardware. Two major goals are accomplished in this stage of development. First, any communication problems between the digital and analog hardware such as scale factors for interface variables, correct assignment of ADC's and DAC's and intolerable amounts of time skewing between analog and digital interface variables are found and corrected. When the Hybrid HWIL real time digital (B digital) is later mated to the analogs for production runs, this same hybrid data interface is incorporated into it, thus eliminating a major error source. Second, the analog hardware is dynamically checked out in closed-loop testing. These dynamic tests are also used after check-out for analog maintenance. In these tests, the digital simulation automatically sets the pots and DCA's to the desired run conditions. Missile pitch-plane only and yaw-plane only tests are made first. These are essentially three degree-of-freedom (DOF) tests in each of the two guidance channels. Then five DOF and finally six DOF tests are completed. The resulting data are compared to overlay plots from similar runs made on the Baseline Digital.

I. SIMULATION CONFIGURATION FOR TRI-FAST

(1) <u>CONFIGURATION FOR THE ALL DIGITAL HWIL (A DIGITAL)</u>. Figure I-1 is a block diagram of the All Digital HWIL simulation showing where the various simulation modules are implemented and a general configuration of the hardware involved. The ACSL simulation comprises the digital component on the CDC 6600 computer. When in real time, the ACSL simulation uses two interrupts with periods of 50 and 10 msec. The slow (50 msec) interrupt segment is used for data logging to the 6600 ECS. The fast (10 msec) interrupt segment contains the interface to the Direct Cell and the digital modules, namely, missile translational and rotational dynamics, autopilot, aero table look-ups, atmosphere, missile rocket motor and actuator. For Tri-FAST, the actuator is a special model that was developed at ASC to run at larger integration step sizes. A modification was made to the autopilots so they could be used at the larger step size. Also included in the fast interrupt segment is a simplified "piggyback" seeker module which contains only the switchable tracking error filter and the integration of estimated inertial line-of-sight rates to obtain gimbal angles. Volume II contains the math model descriptions. The piggyback seeker is used for check-outs with the RFSS hardware that does not include the hardware seeker (see Volume IV, Section 5). For the All Digital HWIL, the only real time digital communication is with the Direct Cell. For Tri-FAST, the 6600 sends the missile acceleration component normal to the seeker antenna, the missile velocity and acceleration vectors in Inertial coordinates and the body rotational rate vector in Body coordinates to the RFSS Datacraft via the Direct Cell; and the Datacraft via the Direct Cell sends seeker line-of-sight rate commands, seeker gimbal angles, target and missile position in Inertial coordinates and seeker range rate to the 6600. The Datacraft computer integrates the missile velocity vector to obtain the missile range in Inertial coordinates. Target range is generated by the Datacraft and compared with missile range to determine the position of the target on the array. The Datacraft combines the missile body rates to form the pitch/yaw/roll angle command sequence that drives the three-axis gimbal system which holds the

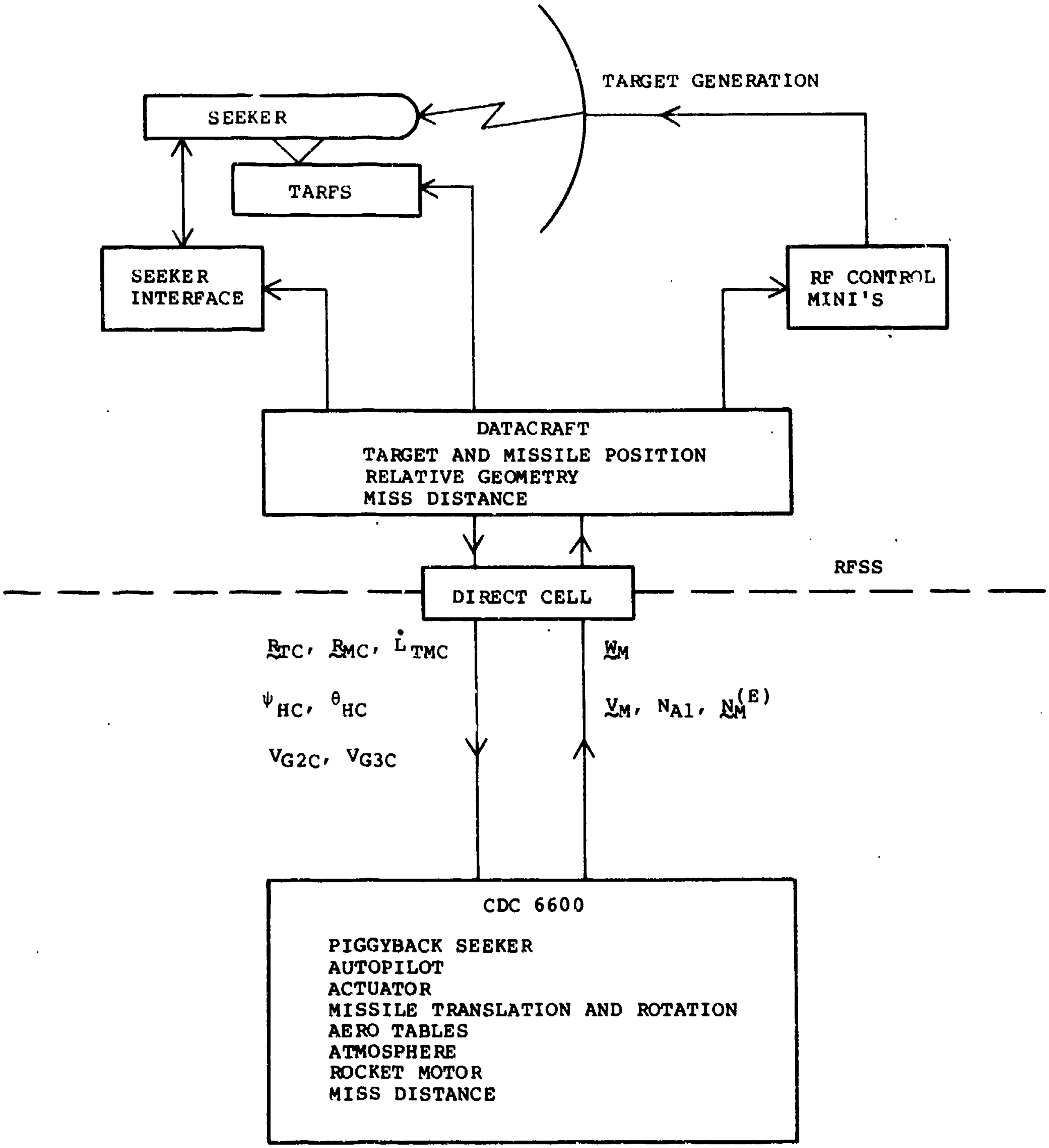


Figure I-1. The All Digital HWIL configuration (A digital).

seeker hardware. Note that the three-axis table orientation is not necessarily the same as that of the missile orientation calculated on the digital because of attitude bias and Euler sequence. Attitude bias is the practice of biasing the initial attitude in the RF chamber to make maximum utilization of the 40-deg array. The missile acceleration normal to the seeker antenna is sent directly to the seeker. The seeker signals are digitized and passed through the Datacraft along with target position to the CDC 6600 via the Direct Cell where they are sampled by the ACSL simulation.

(2) CONFIGURATION FOR THE HYBRID ONLY SIMULATION (CDIGITAL PLUS ANALOG). Figure I-2 is a block diagram of the pure hybrid simulation showing where the various simulation modules are implemented and a general configuration of the hardware involved. The ACSL simulation comprises the digital component on the CDC 6600 computer. When running in real time, the ACSL simulation uses one interrupt with a period of 10 msec. The real time segment for this interrupt contains a seeker module, missile translational dynamics and target motion. The seeker module is even more simplified than the one used in the All Digital HWIL. A description of this seeker model can be found in Volume II. The target relative velocity vector is crossed with the relative target-to-missile range vector to obtain the inertial line-of-sight rate seen by the seeker. The gimbal angles are determined from the target-to-missile unit vector (the direction cosines). The interrupt also contains the interface to the ADC's and DAC's for communication with the analog hardware. The AD-4 analog computer calculates the missile translational accelerations and angles of attack. The AD-4 generates missile velocities in the missile frame, whereas the translational module in the above digital rotates the velocities to the inertial reference frame using the analog-generated Euler angles and integrates them to obtain the inertial missile position. The angles of attack are calculated differently than in the baseline digital to assist in decoupling the higher frequency loops between the analog and digital computers (see Volume III, Subsection 3.A). The EAI 781 analog computer contains the rotational dynamics, actuator and autopilot.

The MVFG performs function generation and contains all the tables for aerodynamic coefficients, atmospheric variables, motor thrust and variables that are a function of changing mass due to motor thrust.

The ACSL digital program uses its seeker and target-to-missile information to generate the seeker line-of-sight

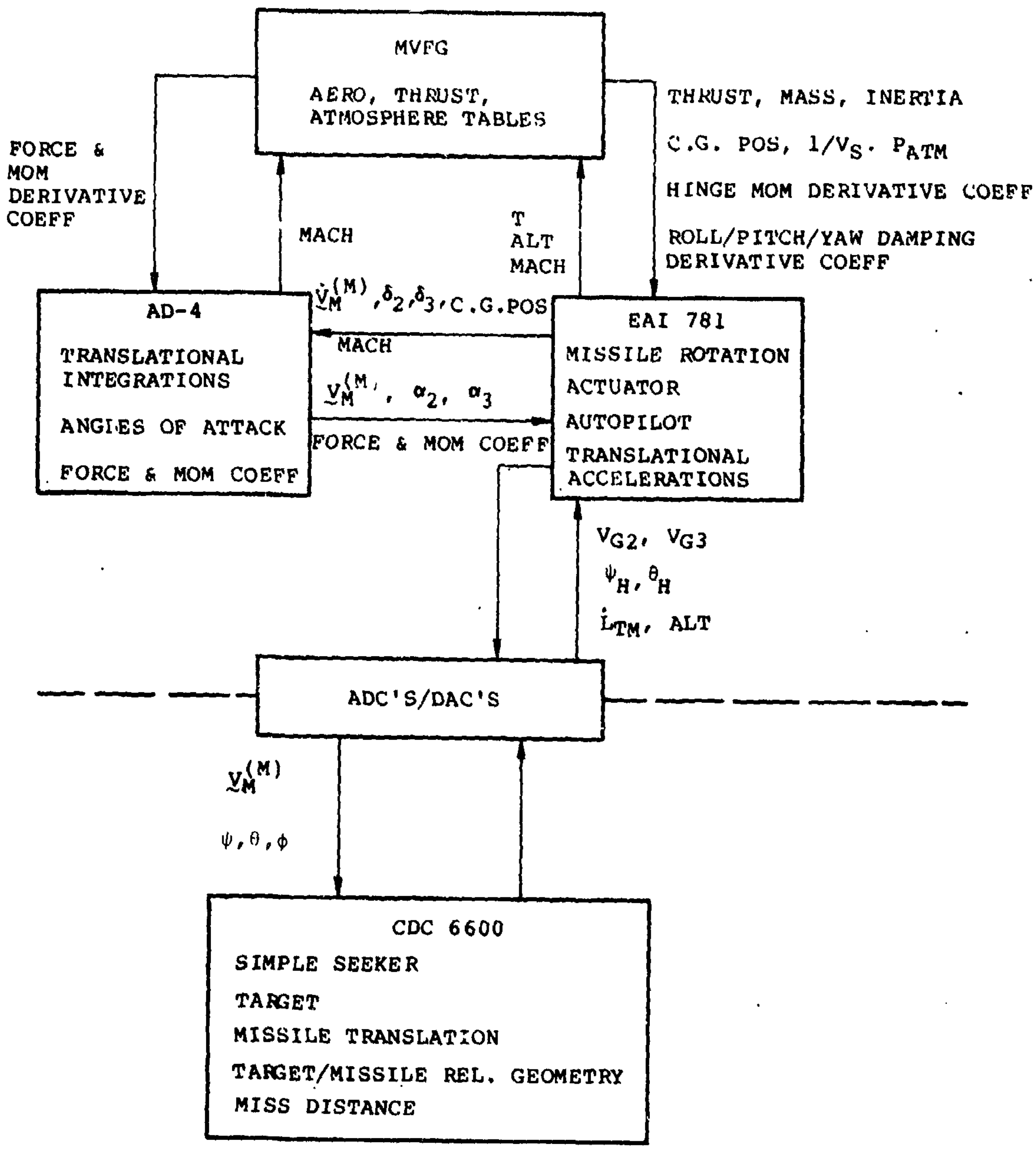


Figure I-2. The Hybrid Only configuration (C digital plus analog).

rate commands, the target-to-missile range rate, the seeker gimbal angles and missile altitude above sea level which are all sent to the EAI 781 via the DAC's. The AD-4 and EAI 781 send the missile velocity vector in the body frame and Euler angles, respectively, to the ACSL program via the ADC's. The digital uses these to find the inertial target-to-missile range rate, which closes the loop for the Hybrid Only simulation.

(3) CONFIGURATION FOR THE HYBRID HWIL (B DIGITAL PLUS ANALOG). Figure I-3 is a block diagram of the Hybrid HWIL simulation showing where the various simulation modules are implemented and a general configuration of the hardware involved. The ACSL simulation comprises the digital component on the CDC 6600 computer. When running in real time, the ACSL simulation uses two interrupts with periods of 50 and 10 msec. The slow (50 msec) interrupt is concerned with data logging to the 6600 ECS. The fast (10 msec) interrupt communicates to the Direct Cell and the ADC's and DAC's for the analog hardware. The missile translational dynamics and a simplified "piggyback" seeker are also in the fast interrupt. The "piggyback" seeker is used in hybrid HWIL interface and check-out and the model itself is the same one used in the pure hybrid simulation (see Volume IV, Section 7). The missile translational model is the one used in the pure hybrid simulation. The analog hardware used and the split of the models among them are discussed in Subsection 3.I, paragraph (2).

For Tri-FAST, the 6600 sends the missile acceleration component normal to the seeker antenna, the missile velocity and acceleration vector components in Inertial coordinates, and the body rotational rate vector components in Missile coordinates to the Direct Cell; and the Datacraft sends the inertial missile and target position to the 6600. So far, this is exactly the same communication linkage as used in the All Digital HWIL. The difference is that the continuous hardware seeker line-of-sight rate commands, gimbal angles, and range rate bypass the Direct Cell and are trunked directly to the analog hardware. The analog hardware sends information to the RFSS only through the 6600. The missile body rotational rates are shipped from the EAI 781 to the RFSS via the Direct Cell and 6600. The missile altitude above sea level and software seeker line-of-sight rate commands, gimbal angles and target-to-missile range rate are sent to the EAI 781 from the 6600. The missile velocity vector in the missile frame, hardware seeker gimbal angles, missile translational acceleration vector in the missile frame, missile Euler angles, missile body rotation rates and

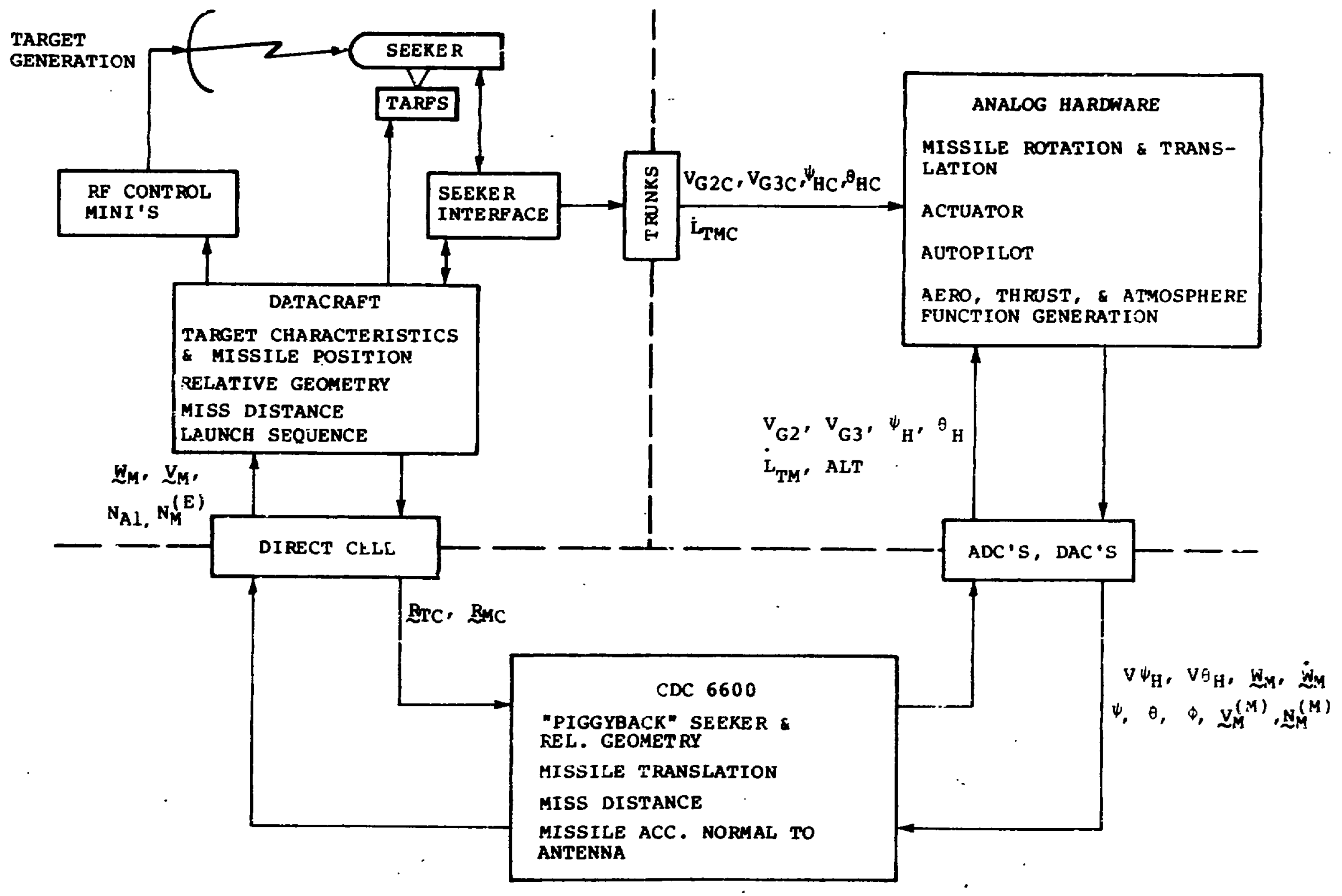


Figure I-3. The Hybrid HWIL configuration (B digital plus analog).

accelerations and C.G. position are sent to the 6600 from the analog hardware. The EAI 781 contains the switches for accepting inputs that are either the hardware or software seeker line-of-sight rate commands, gimbal angles and target-to-missile range rate. The switching is controlled by the 6600. The main changes to the analog/digital interface from the pure hybrid are the additions of missile body rotational rates and accelerations, missile translational accelerations, hardware seeker gimbal angles and missile C.G. position. These are needed by the ACSL digital to calculate the acceleration component normal to the seeker antenna and the software seeker outputs.

J. PRODUCTION CONFIGURATION AND OPERATION SEQUENCE

The man/computer interface for HWIL testing and the operational steps required to make a run or a set of Monte Carlo runs are set forth here. The physical locations of the various devices to be used in testing are discussed. The production configuration is shown in Figure I-4.

The control of the hardware-in-the-loop simulation lies mainly with the RFSS master computer, a Datacraft 6024/1. The master computer controls the minicomputers for RF generation and control. The starting and stopping of the simulation is controlled by the Datacraft. Pertinent peripherals for the Datacraft include:

- A line printer which outputs the engagement termination data variables shown in Figure I-5;

- A Tektronix 4610 hard copy unit which outputs the engagement summary plot (Figure I-6) that was displayed on a Tektronix 613 storage display and the statistical display and printout (Figure I-7) that was displayed on the Tektronix 4012 graphics terminal;

- A Hewlett-Packard 2645A terminal which is available to provide manual control of the Datacraft;

- The weapons system control panel (WSCP) which is a programmable control and display device which among other things controls the scenario (target position and velocity), target type and missile type being run through a set of switches (Figure I-8); and

- The simulation control panel (SCP) which indicates and controls the simulation modes, RESET, RUN, and HOLD (Figure I-9).

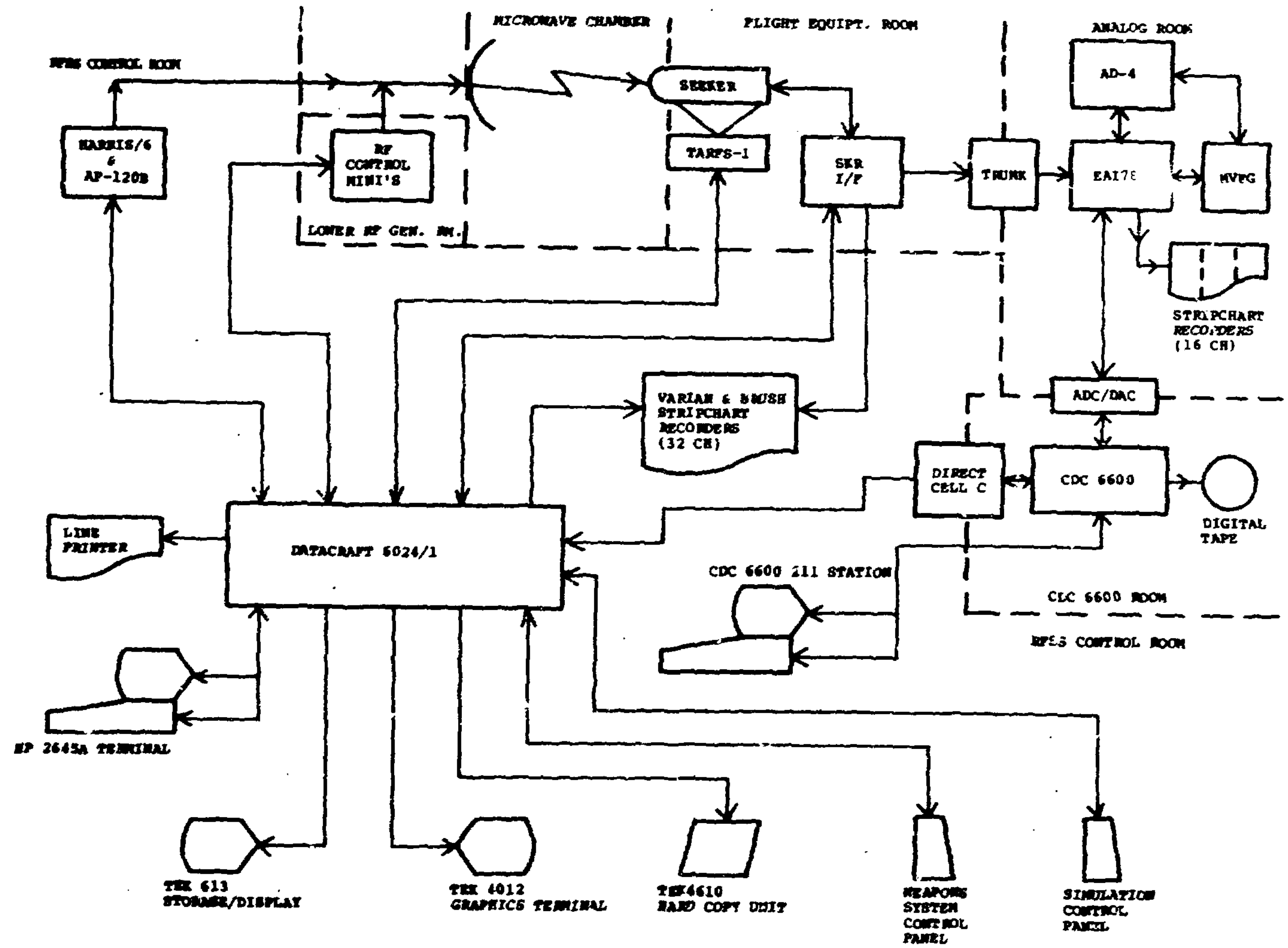


Figure I-4. HWIL simulation production configuration.

VARIABLE NAME AND DEFINITION	COORDINATES	HARDWARE SOURCE	UNITS
BSIRFZ- Az Lab problem bias	Earth	Datacraft	Deg
BSITZ - Random bias in initial target aspect	Target	Datacraft	Deg
BTHRFZ- El Lab problem bias	Earth	Datacraft	Deg
B*1 - Target array limit time		Datacraft	Sec
DL2Z - U/D fin position	Missile	6600/Analog	Deg
DL3Z - R/L fin position	Missile	6600/Analog	Deg
FIMICZ- Initial missile roll angle	Earth	Datacraft	Deg
FIMCZ - Missile roll angle	Earth	Datacraft	Deg
MRMISS- Miss distance		Datacraft	Ft
MVMC - Missile speed		Datacraft	Ft/sec
MVTC - Target speed		Datacraft	Ft/sec
MVTM - Missile-to-target relative speed		Datacraft	Ft/sec
NM1 - Missile longitudinal acceleration	Missile	6600/Analog	Ft/sec^2
NM2 - Missile R/L acceleration (Earth after	Missile	6600/Analog	Ft/sec^2
NM3 - Missile U/D acceleration (Run 979)	Missile	6600/Analog	Ft/sec^2
PSILIZ- R/L Euler angle of line-of-sight vector	Lab	Datacraft	Deg
RM1C - Missile downrange	Earth	Datacraft	Ft
RM2C - Missile cross range	Earth	Datacraft	Ft
RM3C - Missile vertical position	Earth	Datacraft	Ft
RPN2Z - Autopilot U/D proportional navigation (w/o body rate)	Missile	6600/Analog	Deg/sec

Figure I-5. Tri-FAST printout at engagement termination from Datacraft.

VARIABLE NAME AND DEFINITION	COORDINATES	HARDWARE SOURCE	UNITS
RPN3Z - Autopilot R/L proportional navigation (w/o body rate)	Missile	6600/Analog	Deg/sec
RSL3 - Missile launch altitude above sea level	Earth	Datacraft	Ft
RTIC1 - Target downrange position at missile launch	Earth	Datacraft	Ft
RTIC2 - Target cross range position at missile launch	Earth	Datacraft	Ft
RTIC3 - Target altitude AGL at missile launch	Earth	Datacraft	Ft
RT1C - Target downrange	Earth	Datacraft	Ft
RT2C - Target cross range	Earth	Datacraft	Ft
RT3C - Target vertical AGL (S/A) or SL (A/A)	Earth	Datacraft	Ft
SIMICZ - Az Euler angle at missile launch	Earth	Datacraft	Deg
SIMTZ (ASP3C) - R/L missile from target	Target	Datacraft	Deg
SIMZC - Missile az Euler angle	Earth	Datacraft	Deg
SITC - Target az Euler angle	Earth	Datacraft	Deg
TFLTMX - Flight termination time		Datacraft	Sec
THMICZ - El Euler angle at missile launch	Earth	Datacraft	Deg
THMTZ - U /D missile from target	Target	Datacraft	Deg
THMZC - Missile el Euler angle	Earth	Datacraft	Deg
THTLIZ - U/D Euler angle of line-of-sight vector	Lab	Datacraft	Deg
TNTRCP - Projected closest miss time			Sec

Figure I-5.

VARIABLE NAME AND DEFINITION		COORDINATES	HARDWARE SOURCE	UNITS
VM1	- Missile downrange rate	Earth	6600	Ft/sec
VM2	- Missile cross range rate	Earth	6600	Ft/sec
VM3	- Missile vertical rate	Earth	6600	Ft/sec
VRT1	- Target downrange rate relative to missile	Earth	Datacraft	Ft/sec
VRT2	- Target cross range rate relative to missile	Earth	Datacraft	Ft/sec
VRT3	- Target vertical rate relative to missile	Earth	Datacraft	Ft/sec
VT1C	- Target downrange rate	Earth	Datacraft	Ft/sec
VT2C	- Target cross range rate	Earth	Datacraft	Ft/sec
VT3C	- Target vertical rate	Earth	Datacraft	Ft/sec
WMZ1	- Missile roll rate	Missile	6600/Analog	Deg/sec
WMZ2	- Missile U/D rate	Missile	6600/Analog	Deg/sec
WMZ3	- Missile R/L rate	Missile	6600/Analog	Deg/sec
XMISST-	Closest miss X-component	Target	Datacraft	Ft
YMISST-	Closest miss Y-component	Target	Datacraft	Ft
ZMISST-	Closest miss Z-component	Target	Datacraft	Ft

NOTE:

1. Adding suffix Z means the variable is in degrees, g's, oz-in as opposed to radians, ft/sec^2, or ft-lb.

2. The target model no., missile configuration designation, and scenario no. was printed out.

3. Adding suffix C means the variable value is that from the RFSS rather than the CDC 6600.

4. 6600 variables are also defined in Appendix D. The variables will not contain the Z and C suffixes.

Figure I-5. (Concluded)

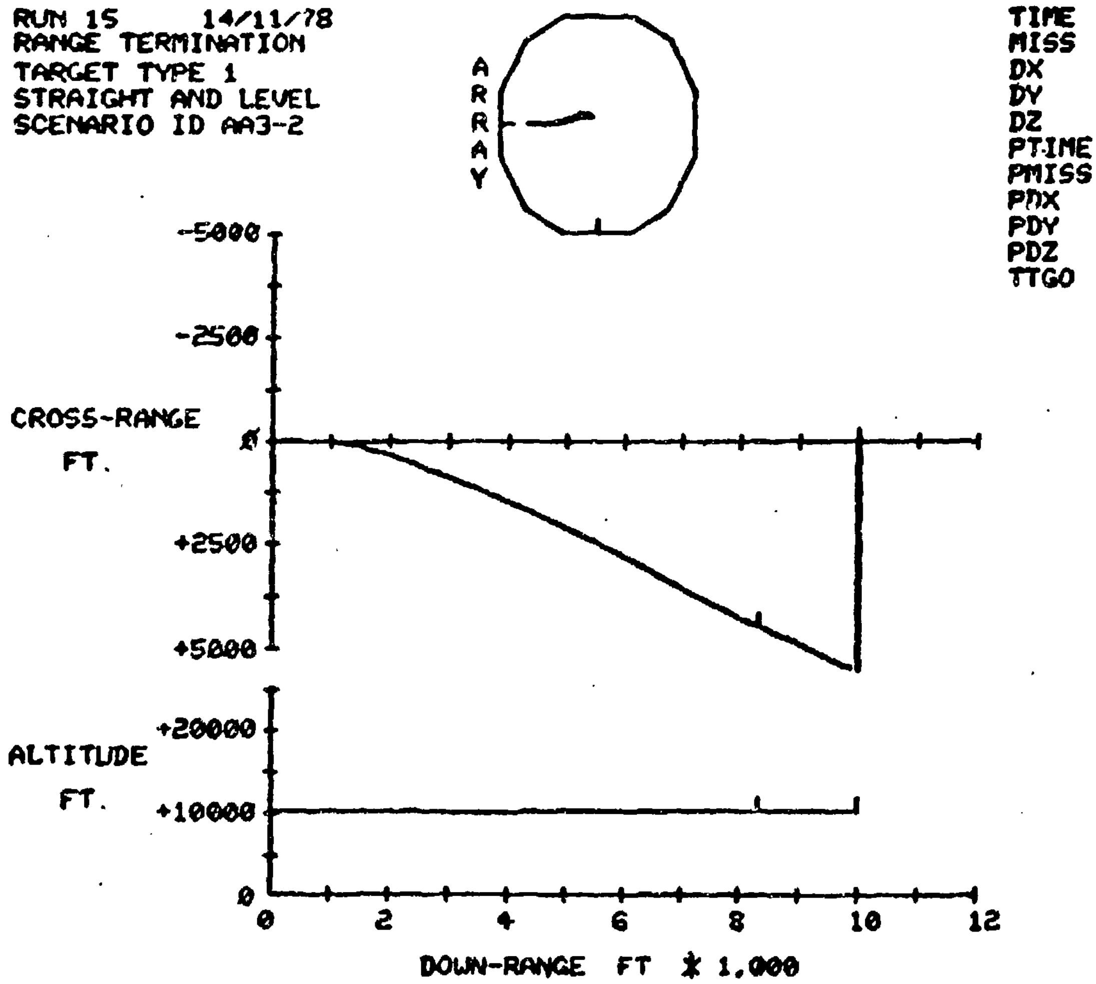


Figure I-6. Sample Tektronix 4610 hard copy from Datacraft.

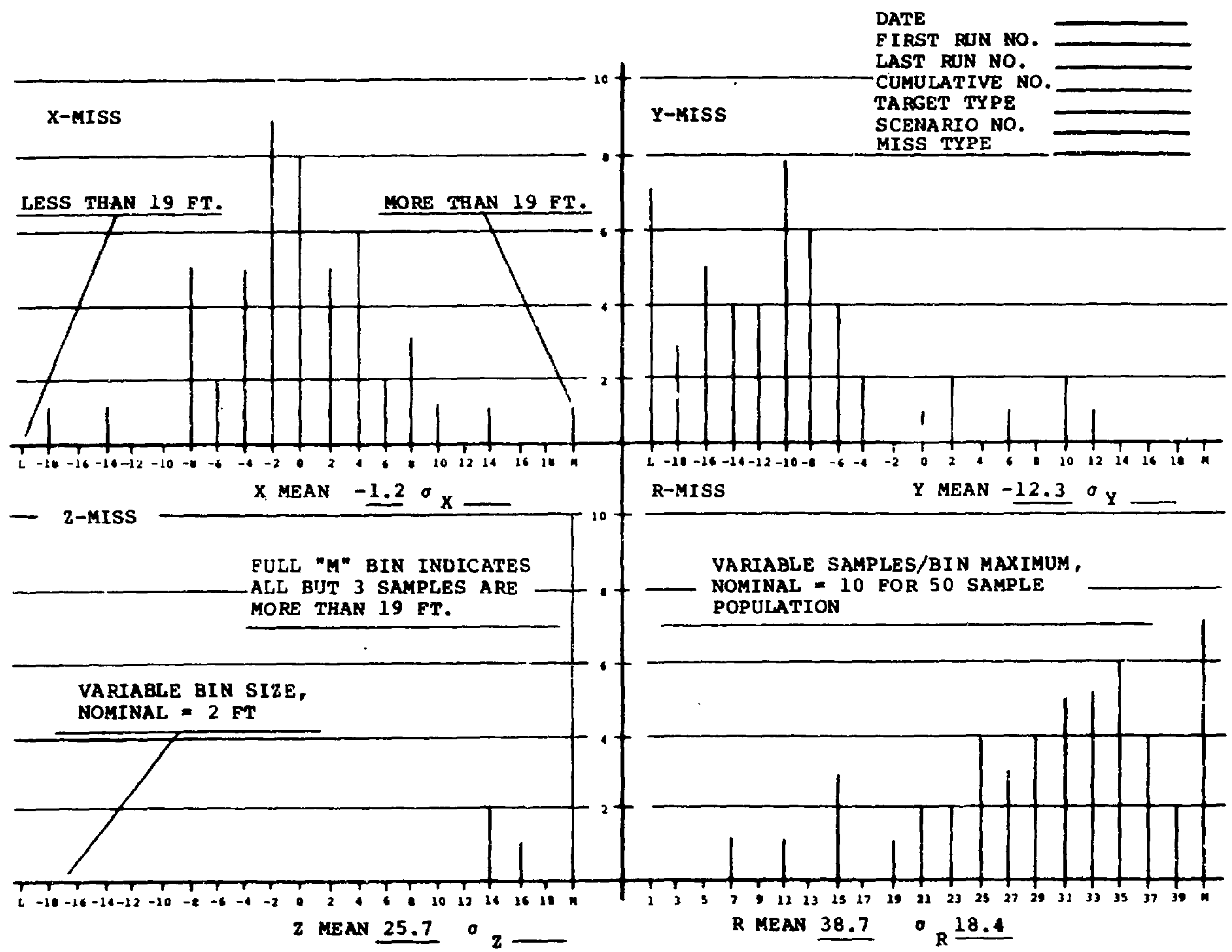


Figure I-7. Typical RFSS Tri-FAST statistical display and printout.

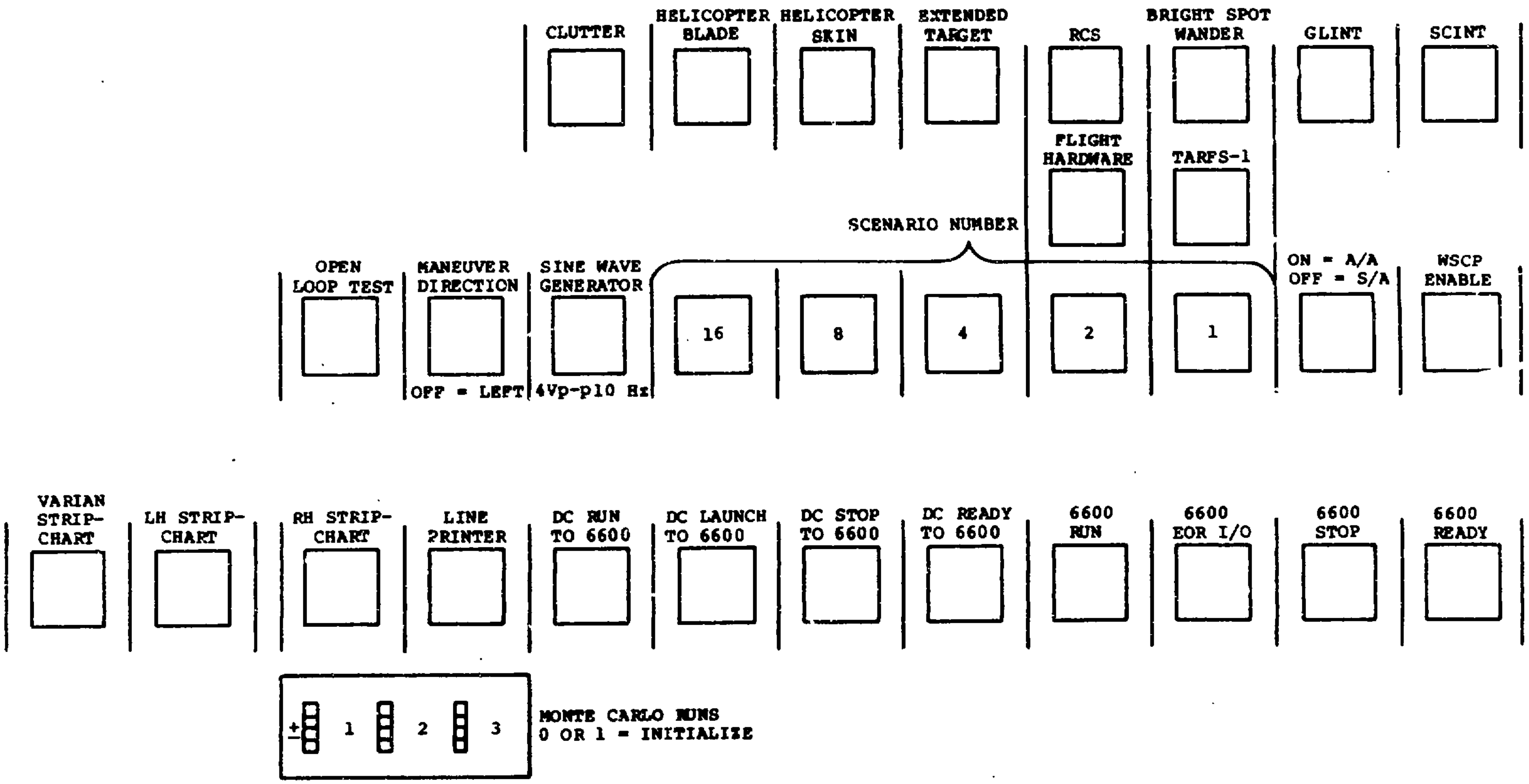


Figure I-8. Weapons system control panel push buttons and thumb wheels for controlling target type, scenario number, missile type and the number of Monte Carlo runs.

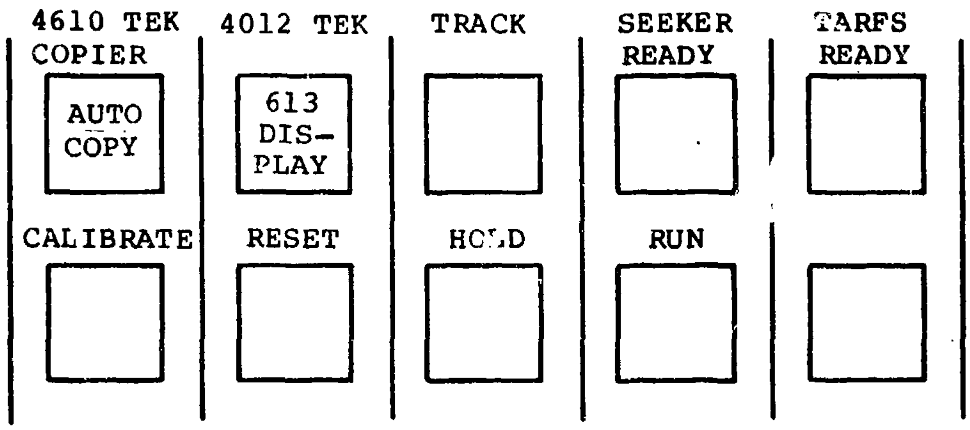


Figure I-9. Simulation control panel push buttons and indicator lights.

There are several other devices interfaced with the Datacraft but they are not as germane to the user. The Datacraft main frame and peripherals are located in the RFSS control room. The interrupt for communication between the Datacraft, CDC 6600 and analog hardware is an external interrupt generated by a Systron Donner Digital Time Interval Generator which is located in the RFSS control room.

Also located in the control room are two 8-channel Brush ink stripchart recorders and a 16-channel Varian electrostatic recorder. Appendix P lists the variables to be recorded on these devices.

The CDC 6600 controls the hybrid portion of the hardware-in-the-loop simulation. The major peripheral for the 6600 is a CDC 211 Entry/Display Station that has available to it a special software package called the Digital Display System (DDS). The DDS software is particularly suitable for controlling and monitoring the real time digital portion of the hybrid simulation. The 211 is a CRT with keyboard. This hardware will be referred to as the DDS console and is physically located in the RFSS control room. ACSL run time commands can be inserted into the 6600 via the DDS console.

The bulk of the simulation resides on the EAI 781 and AD-4 analog computers and the MVFG. The different missile configurations were controlled on the analog portion of the hybrid simulation by changing pot and DCA settings and inserting new tables into the MVFG. Two 8-channel Brush ink stripchart recorders were available for recording variables from the analog computers as specified in Appendix P. The analog hardware and associated recorders are located in the ASC analog room.

The following constitutes a sketch of the operational sequence for a HWIL simulation. The Datacraft is put into the RESET mode by manually pushing a button on the RFSS panel (Figure I-9) and the Datacraft automatically attenuates the RF array target signal to a minimum. This causes the seeker to go into its SEARCH mode. The scenario for this simulation run is manually set by switches (target maneuver, position, velocity, etc.) on the RFSS weapons system control panel (Figure I-8) and the Datacraft automatically calculates the TARFS-1 gimbal angles for missile launch, commands the TARFS to move the seeker body to its proper attitude, calculates the initial target from missile line-of-sight angles from which it derives the initial seeker gimbal angles, and finally commands the seeker head to its proper position. Once the seeker gimbal angles and TARFS angles are in tolerance (lights on the RFSS simulation control panel go on), the run button on the simulation control panel is pressed and the Datacraft tells the 6600 that it is ready. Parallel to this operation, the 6600 via the DDS console, is also being set up.

The DDS console operator "attaches" the file containing the real time job statements to his console and instructs the 6600 computer to execute the file. These actions cause the absolute file containing the ACSL real time digital portion (which in itself contains a real time and a batch background section) of the simulation to be executed. A major difference between a real time job and a batch job is that the real time job occupies a control point in central memory and cannot be rolled in and out of central memory as a batch job can. Thus, the real time program is always available for execution by the central processor unit (CPU).

The operator then enters the ACSL run time commands (or the 6600 reads a run time command file) for setting parameter and initial condition (I.C.) values and for starting up the digital batch background portion of the simulation. The batch background reserves the interrupts, Direct Cell C, ADC's and DAC's for this job. The batch background strobes Direct Cell C every 10 msec to check if the Datacraft is ready to transmit data (as discussed above). When the Datacraft is ready, the initial missile Euler angles and position vector and the initial target position vector are read from Cell C and unscaled by the batch background program.

The Analog Controller, which is a hardware/software system that enables the 6600 to monitor and control the analog machines, is reserved. The initial conditions on the analogs that are to be set by the 6600 are the missile body rate vector, missile speed and the missile Euler angles. The corresponding analog hardware to be set are DCA's on the EAI 781 and servo set pots on the AD-4. The 6600 scales and sets these initial conditions on the analog. The Analog Controller is then returned to the 6600 system. The strip-chart recorders in the analog room are turned on by the 6600 by "jamming" (information is forced across a DAC without the firing of any interrupts) a discrete to the 781.

The batch background now asks the 6600 for permission to enter real time. Equipment reservations are made. The 6600 static scheduler checks the required compute time per interrupt frame for this job against the present usage of the 6600 CPU by other real time jobs. The job then enters the real time mode. Each interrupt is fired once using a 6600-generated software interrupt. The 6600 tells the Datacraft it is in real time. No integrations take place on the 6600. Only the initial interrupt communications among the Datacraft, 6600 and analog computers are accomplished. The simulation interrupts are then driven by a RFSS external interrupt for the rest of the simulation run. The 6600-generated interrupt for data recording in ECS also starts firing. The batch background part of the program is put into the hold mode.

Once the 6600 says it is in real time, the Datacraft brings up the RF minicomputers and consequently the target and starts integrating target velocity to produce RF target motion via the RF minicomputers. Before it does so, it tells the 6600 that the target is about to start moving. When the 6600 receives this message from the Datacraft, it starts reading missile and target position from the Datacraft. The seeker acquires the target in range and Doppler and then starts tracking in phase and angle. If the seeker goes into track mode before the RF target reaches the point where missile launch is to occur, the Datacraft sets the LAUNCH MISSILE variable bit high and sends it to the Direct Cell in a discrete word. On the next firing of the interrupt, the 6600 sees the LAUNCH MISSILE bit is high and it sets the time of launch to the present time in the digital portion of the simulation and starts integrating for the missile states. Correspondingly, the 6600 sets a bit in a DAC discrete word high and on the next firing of the interrupt the analog portion of the hybrid simulation fires the missile rocket motor. The EAI 781 transmits missile translational and rotational motion data to the 6600 via the ADC's, which in turn sends it to the Datacraft via the Direct Cell. The Datacraft resolves the missile body rates into Euler angle rates and integrates for the Euler angles. The Euler angles are transformed to the pitch/yaw/roll angle commands that are sent to the TARFS to produce a rotational motion similar to that the seeker would see in actual flight. Missile velocity is integrated for missile position which is combined with target position to determine the position of the RF source(s) on the RFSS antenna array. Missile position, velocity and acceleration are used to calculate the range delay and Doppler on the RF target signal emanating from the array. The loop is closed through the hardware seeker by its responding to changes in the RF target in the form of guidance commands that are trunked directly to the simulated autopilot on the EAI 781.

When the time to go before missile intercept of the target is less than 10 msec, or the TARFS' gimbals angle limit, or the RF source moves off the array, etc., the Datacraft calculates miss distance and tells the 6600 to stop. The 6600 sends a similar message to the analog computers telling them to stop. The 6600 is taken out of real time in the interrupt for math model calculations.

When the real time portion of the 6600 digital program stops, the batch background is activated. The batch background buffers data from ECS to a mass storage file. The 6600 tells the Datacraft it has ended the hybrid portion of this simulation run. When the Datacraft receives this message, it sends a special buffer to the Direct Cell containing the end of run data. After this data buffer is filled, the Datacraft sends the 6600 a message saying the Datacraft has ended its portion of the run and the special end of run data are ready to be read. The 6600 batch background program strobes the Direct Cell every 10 msec to see if the Datacraft has sent this message. Upon its reception, the batch background reads, among other things, the missile and target position from the Direct Cell and uses this information in its miss distance calculation. The real time hardware is unreserved. The run number and Datacraft-calculated miss distance are also part of the end of run data (Appendix F or P). The end of run data are put on a special file and the mass storage file containing the data dumped from ECS is added to the end of run data (see Volume III, Subsection 2.C). If a Monte Carlo set of runs is being made, the Datacraft indicates this to the 6600 through the end of run data. The 6600 batch background then immediately sets up for another real time run without accepting any ACSL run time commands. For just one run the normal program flow is to allow the ACSL executive to process run time commands before real time set-up. Meanwhile, the Datacraft is returned to the RESET mode awaiting inputs from the weapons system control panel.

4. HARDWARE-IN-THE-LOOP PRODUCTION TESTING

Figure I-10 lists the air-to-air (A/A) and surface-to-air (S/A) scenarios actually tested. The scenario launch conditions are specified in Appendix O. Versions A and C (Appendix N), the "parameters document" and "tactical" versions, of both the A/A and S/A missiles were tested. RF target types I, II, III, and V (Subsection 4.A) were used.

The Real Time Digital HWIL simulation was used for most of the testing. For production testing only the A/A range switch value was used for switching the gain of the seeker tracking loop filter. The S/A autopilot did not use attitude biasing. Missile roll angle at launch was -45 deg for both the A/A and S/A missiles. Because the update rate on the hardware seeker range rate was relatively low, the software-generated range rate was used in the autopilot gain switching for both the Real Time Digital/RFSS and Real Time Hybrid/RFSS simulations.

A. TRI-FAST RF TARGET MODELS

The development of five target models, described below, was required for Tri-FAST.

(1) DEFINITIONS. In the descriptions of the target models, the following terms are used:

- Deterministic RCS variation: Low frequency, deterministic variation with aspect to target RCS.

- Scintillation: Wide bandwidth statistical variation with time of target RCS. The statistical parameters such as mean, variance and bandwidth may be aspect or aspect rate dependent. The aspect dependent mean scintillation corresponds to deterministic RCS variation.

- Bright Spct Wander: Low frequency, deterministic variation with aspect to target scattering center location.

	TARGET TYPE					VERSION		SIMULATION		ROLL	
SCENARIO	I	II	III	IV	V	A	C	DIGITAL	HYBRID	IN	OUT
A/A2	X					X		X		X	X
		X				X		X			X
			X			X		X			X
A/A3	X					X		X		X	X
	X								X		X
	X						X	X			X
	X						X		X		X
		X				X		X		X	X
			X			X		X			X
			X			X			X		X
			X				X	X			X
			X				X		X		X
A/A4	X					X		X		X	X
		X				X		X			X
			X			X		X			X
A/A7	X					X		X			X
	X					X			X	X	
		X				X		X			X
			X			X		X			X
S/A2	X					X		X		X	X
		X				X		X			X
			X			X		X			X
S/A5	X					X		X		X	X
		X				X		X			X
			X			X		X		X	X
S/A8	X					X		X		X	X
	X					X			X		X
					X	X			X		X
S/A9	X					X		X			X
	X					X			X		X
					X	X			X		X
S/A10	X					X		X		X	
		X				X		X			X
			X			X		X			X
S/A14	X					X		X		X	X
	X						X		X		X
		X				X		X			X
			X			X		X			X
			X				X		X		X

Figure I-10. HWIL simulation tests completed.

• Glint: Wide bandwidth statistical variation with time of target scattering center location. The statistical parameters such as mean, variance and bandwidth may be aspect or aspect rate dependent. The aspect dependent mean glint corresponds to bright spot wander.

(2) TARGET MODEL DESCRIPTIONS

(a) TARGET MODEL I: POINT TARGET. This is an isotropic scatterer localized in space at the physical target C.G. This target exhibits no glint, scintillation, bright spot wander or deterministic RCS variation. The value of the fixed RCS will be selected prior to each block of runs.

(b) TARGET MODEL II: EMPIRICAL TARGET. This is a scatterer, localized in space at any instant of time, which exhibits deterministic RCS variation and bright spot wander derived from smoothed, empirical data. The RCS variation and bright spot wander versus aspect data were supplied by Naval Weapons Center.

(c) TARGET MODEL III: EMPIRICAL/STATISTICAL TARGET. Model III is a scatterer, localized in space at any instant of time, which exhibits glint and scintillation including the smoothed, empirical target characterizations of Target Model II. The statistical parameters of the fluctuations such as mean, variance and bandwidth will be aspect or aspect rate dependent. The aspect dependences of the means and variances will be derived from empirical data and will be consistent with the data supplied for Target Model II. These data were supplied by Naval Weapons Center.

(d) TARGET MODEL IV: DETERMINISTIC MULTIPLE SCATTERER TARGET. In this model there is a complex target consisting of a cluster of independent, spatially localized scatterers. Each scatterer is described by an aspect dependent location and an aspect dependent RCS, both specified in Target coordinates. The RF implementation of this target model employs a multiple tap delay line to provide range extent. Angle extent is obtained by transmitting each tap signal from independent array locations. The aspect dependent data will be based on empirical data and will be consistent with Target Models II and III. These data were supplied by Naval Weapons Center.

(e) TARGET MODEL V: HELICOPTER. Target Model V is a scatterer, localized in space at the physical target C.G., which exhibits an aspect dependent skin return derived from empirical data and an aspect dependent blade return based on a theoretical blade modulation model. The skin return data and the blade characterization data have been determined by MIRADCOM.

(3) STATISTICAL TARGET MODEL IMPLEMENTATION. A discussion of the statistical target model is presented in two NWC memoranda.[1,2] A summary of the deviations from the referenced model to affect its implementation in the RFSS closed-loop program will be given here. The most general statistical model is a point source with bright spot wander (BSW), statistical glint and Rayleigh process scintillation. The individual components of the model can be enabled or disabled to provide submodels utilizing the various combinations of BSW, average RCS, random glint and scintillation.

The target is assumed to fly in a wings-level attitude with zero angles of attack and sideslip. Line-of-sight angles are shown in Figure I-11. The azimuth (ψ_{LOS} or ASP3C on mag tape) is measured in the target X_T-Y_T plane positive CW from the X_T axis to the projection of the LOS onto the X_TY_T plane. The elevation (θ_{LOS} or ASP2C on mag tape) measured (positive up) from the LOS projection in the X_TY_T plane to the LOS. The aspect angle (α_{LOS}) is the angle between the X_T axis and the LOS.

(4) BRIGHT SPOT WANDER. BSW is obtained (in feet) in the program by table look-up as a function of aspect angle, αLOS. Details of the BSW geometry, sign conventions, and the data table are given in the references previously cited. The apparent shift in target location is assumed to be in the wing plane (X_TY_T) along a line perpendicular to the line-of-sight. Due to the wings-level attitude of the target, this line is parallel to the Y_{LOS} axis (Figure I-12) of the LOS coordinate system.

BSW is implemented in a manner which is software compatible with the implementation of the angular perturbations required by the Extended Target Model; i.e., the BSW value is converted to angle perturbations that are applied directly to the array azimuth and elevation location of the target C.G. in the Lab coordinate system. Assume that the

1. "MIRADCOM Target Models," Department of Navy (NWC) Memorandum 3911/MLM: Cam, Reg. 3911-53-78, dated 22 May 1978.

2. "Corrections and Clarifications to MIRADCOM Target Models," Department of Navy (NWC) Memorandum 3911/MLM: Cam, Reg. 3911-56-73, dated 12 June 1978.

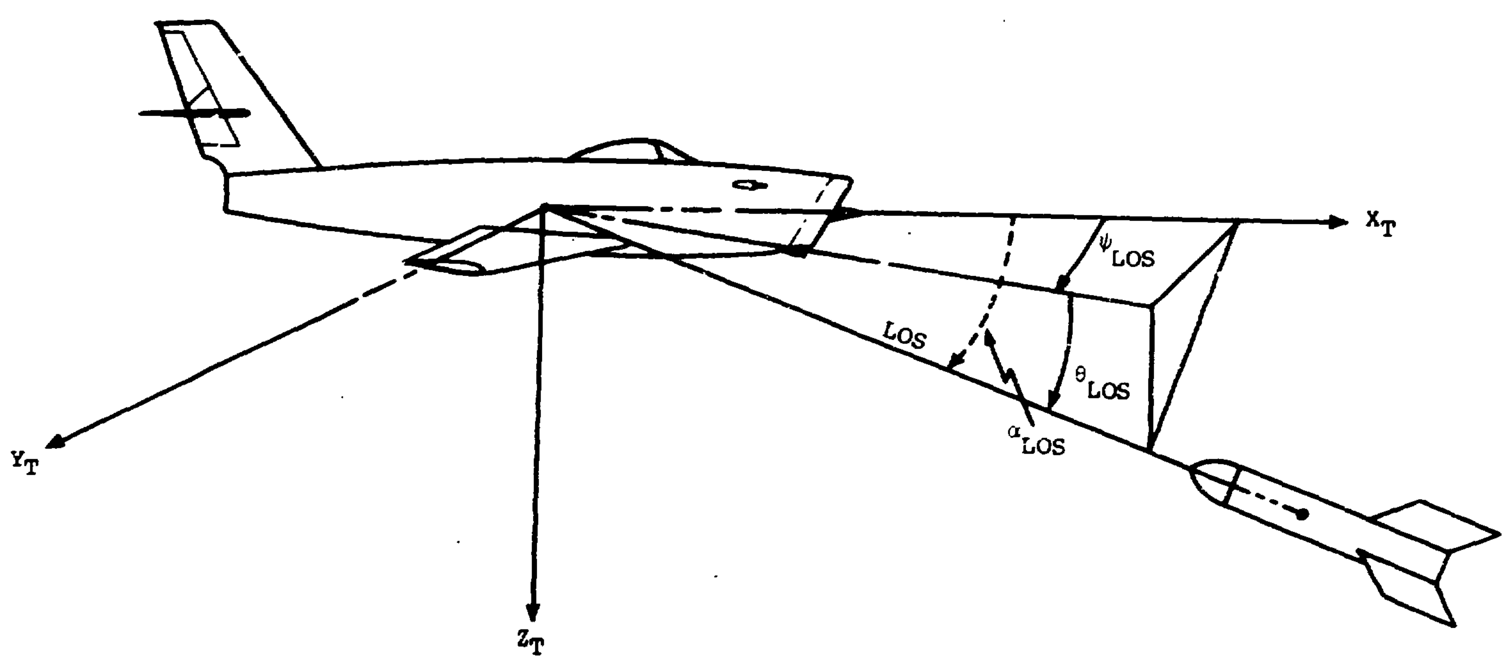


Figure I-11. Line-of-sight angles in Target coordinate system.

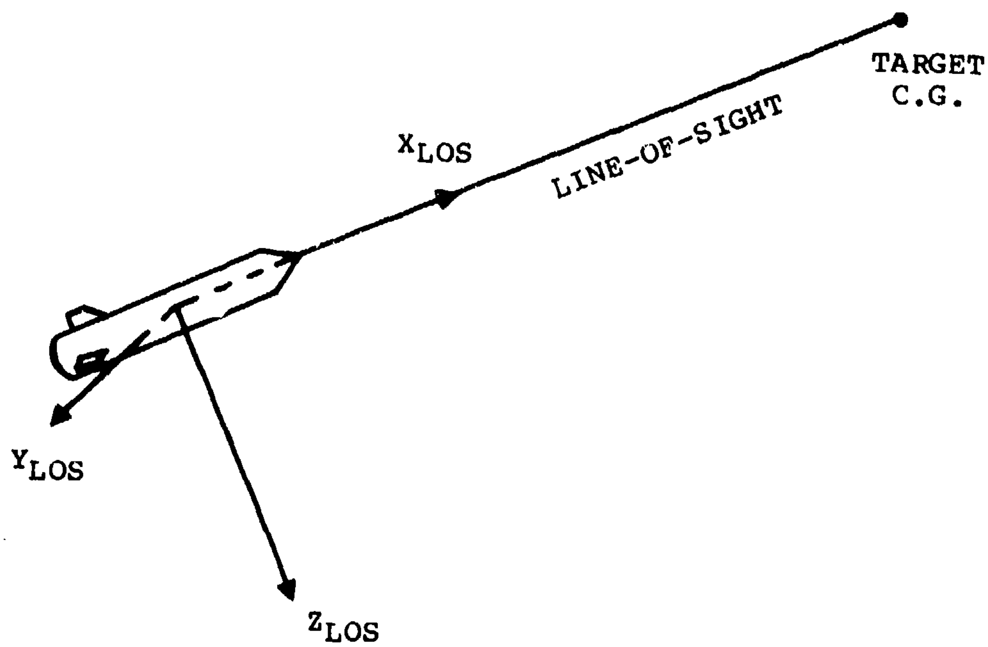


Figure I-12. Line-of-sight coordinates.

unperturbed location is at (ψ_L, θ_L) where ψ_L is the azimuth to the true line-of-sight in Lab coordinates and θ_L is the elevation. The transformation from Lab to LOS coordinates is defined by the angles ψ_L and θ_L plus a rotation ϕ_L about the line-of-sight (Figure I-13). The BSW value is along Y_{LOS} and has orthogonal components

$$Y_{BSW} = BSW * \cos \phi_L$$

$$Z_{BSW} = BSW * \sin \phi_L$$

with Y_{BSW} parallel to the Lab coordinate X_L-Y_L plane. With R, the missile-target range, the azimuth and elevation perturbations $\Delta\psi$ and $\Delta\theta$ are given by

$$\Delta\psi = \frac{Y_{BSW}}{R} \text{ radians}$$

$$\Delta\theta = \frac{-Z_{BSW}}{R} \text{ radians}$$

where the small angle approximation $\tan \alpha = \alpha$ has been applied. This approximation is justified by noting that $|BSW| \leq 2.8$ m and the minimum range $R \geq 28$ m. Thus,

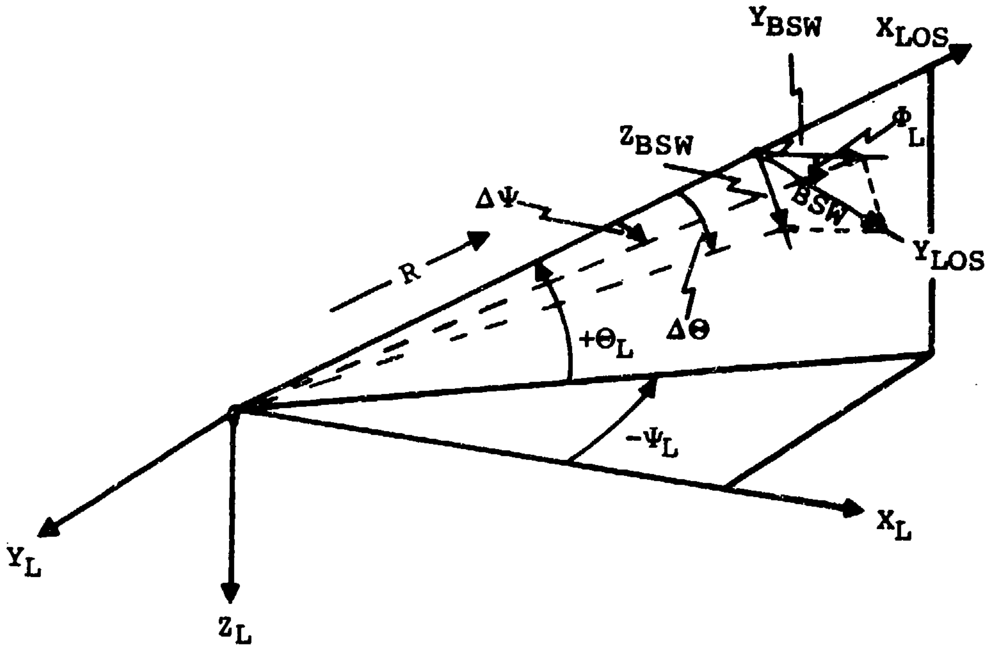


Figure I-13. Euler sequence from Lab to LOS coordinates.

$$\frac{|BSW|}{R} \leq 0.1$$

and

$$\tan^{-1}(0.1) = 0.0997 \text{ radian.}$$

The angles for array control are calculated as follows:

$$\psi_A = \psi_L + \Delta\psi$$

$$\theta_A = \theta_L + \Delta\theta \quad .$$

It should be noted that the angle ψ_A is commanded in the X_L-Y_L plane while the $\Delta\psi$ portion of ψ_A is calculated in a plane inclined by the angle θ_L with respect to X_L-Y_L. This causes some foreshortening of $\Delta\psi$ as a function of θ_L. The maximum error is obtained when $|\theta_L|$ is maximum, i.e., when $|\theta_L| \cong 21$ deg, the half angle of the array. At these angles, the actual angle is

$$\Delta\psi_A = \Delta\psi \cos 21° = 0.93\ \Delta\psi$$

Thus, the worst case error is 7%.

(5) GLINT IMPLEMENTATION. Glint is implemented in the RFSS in a manner analogous to the BSW implementation, i.e., as azimuth and elevation perturbations in the Lab coordinate system. The glint model presented in the references previously cited is a three-dimensional spatial model. This model is reduced to a two-dimensional angular model for RFSS implementation.

Let σ_x, σ_y, σ_z denote the glint 1-sigma values in the Target coordinate system and let $[t_{ij}]$ denote the target-to-LOS transformation matrix. Then the glint 1-sigma values perpendicular to the line-of-sight are as follows:

$$\sigma_y{}^* = [(\sigma_x t_{21})^2 + (\sigma_y t_{22})^2 + (\sigma_z t_{23})^2]^{1/2}$$

$$\sigma_z{}^* = [(\sigma_x t_{31})^2 + (\sigma_y t_{32})^2 + (\sigma_z t_{33})^2]^{1/2} \quad .$$

$\sigma_y{}^*$ is along Y_{LOS} of the LOS coordinate system and $\sigma_z{}^*$ is along Z_{LOS}. These are converted to angular dimensions in the same manner as BSW,

$$\sigma_y{}' = \frac{\sigma_y{}^*}{R} \text{ radians}$$

$$\sigma_z{}' = \frac{\sigma_z{}^*}{R} \text{ radians.}$$

The corresponding glint algorithm outputs are rotated through the angle ϕ_L (see section on BSW implementation) before application to the array angles.

The line-of-sight rotation rates that are required for filter bandwidth calculation are calculated as follows. Let ψ,θ denote the line-of-sight azimuth and elevation in the Target coordinate system; then from Figure I-14,

$$\omega_V = \dot{\theta}$$

$$\omega_W = \dot{\psi} \cos \theta$$

$$\omega_U = -\dot{\psi} \sin \theta$$

$$|W| = (\dot{\theta}^2 + \dot{\psi}^2)^{1/2} \quad .$$

If $(\Delta X_T, \Delta Y_T, \Delta Z_T)$ is the missile position in Target coordinates, then

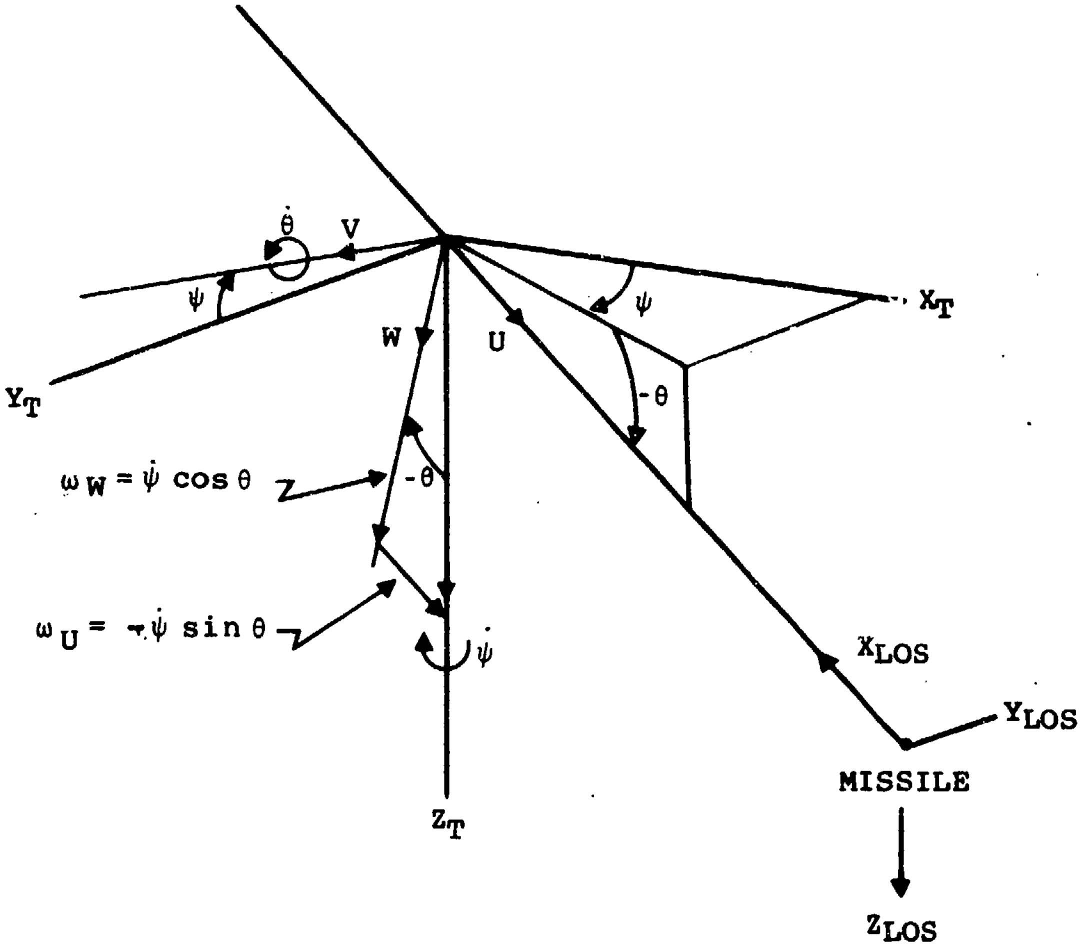


Figure I-14. LOS motion in Target coordinates.

$$\sin\psi = \frac{\Delta Y_T}{R_{xy}} \qquad R_{xy} = (\Delta X_T^2 + \Delta Y_T^2)^{1/2}$$

$$\cos\psi = \frac{\Delta X_T}{R_{xy}}$$

$$\sin\theta = \frac{-\Delta Z_T}{R} \qquad R = (\Delta X_T^2 + \Delta Y_T^2 + \Delta Z_T^2)^{1/2}$$

$$\cos\theta = \frac{R_{xy}}{R}$$

$$\dot{R}_{xy} = \Delta\dot{X}_T\cos\psi + \Delta\dot{Y}_T\sin\psi$$

$$\dot{\psi} = -(\Delta \dot{X}_T \sin \psi - \Delta \dot{Y}_T \cos \psi)/R_{xy}$$

$$\dot{\theta} = -(\dot{R}_{xy} \sin \theta + \Delta \dot{Z}_T \cos \theta)/R \quad .$$

The bandwidths are calculated in Hertz as

$$f_y = 7 * 57.296 * |W_W| \qquad (\sigma_y' \text{ bandwidth})$$

$$f_z = 7 * 57.296 * |W_V| \qquad (\sigma_z' \text{ bandwidth})$$

$$f_s = 7 * 57.296 * |W| \qquad (\text{Fading bandwidth}).$$

A maximum allowable bandwidth of 50 Hz is imposed. This is one-half the desired maximum stated in the references, but is required by the simulation step size to avoid frequency folding.

(6) FADING IMPLEMENTATION. Fading is implemented exactly as specified in the references except for the maximum bandwidth change noted immediately above.

B. HWIL SIMULATION OUTPUTS AND DATA RECORDING

The simulation outputs are as follows:

- Stripcharts
- Digital magnetic tapes
- PCM tapes
- Video tapes
- Datacraft hard copy plots for engagement summary
- Datacraft line printer listings for engagement summary
- ACSL Postprocessor hard copy plots for Monte Carlo data.

Figure I-4 shows the physical location of most of the recording devices. Not shown are the 8-channel Brush stripchart recorder, PCM tape recorder, video cameras and video recorder which were located in the flight equipment room. One video camera was positioned in the microwave chamber to record the movement of the seeker. The other camera recorded the NWC clutter map display. The Brush stripouts and PCM recordings were monitored by and recorded specifically for the seeker contractor, Motorola.

The RFSS Control Room contained 16 channels each of Brush stripcharts and Varian stripcharts. Appendix P lists the variables and scaling written on these recorders. The variables on the Brush machines changed but those listed were used the most often. Appendix P also contains the variables and scaling used on the 16 channels of Brush recordings in the Analog Room. Figure I-5 lists the variables recorded on the Datacraft line printer. Figure I-6 is an example of the Tektronix 4610 hard copy plots produced through the Datacraft. The generation of digital magnetic tapes by the CDC 6600 is discussed in Subsection 4.C. Subsection 4.D contains a discussion on the ACSL Postprocessor hard copy plots produced on a Tektronix 4610.

The coordinate systems for the output variables are defined in Volume II, Section 2 except for those specified in RFSS reference frames. RFSS reference frames are discussed in Subsection 4.E.

C. CDC 6600-GENERATED MAGNETIC TAPES

The Tri-FAST Data Tapes contain coded data in ACSII format for Motorola or "Fieldata" format for NWC in physical records of 1200 characters each. The data for each flight is described by a single header record followed by one record for each sample time point, followed by an end of file. All numeric data in each record are contained in fields, 12 characters wide.

The format of the header record is as follows:

- 120 characters of Hollerith coded title
- 28 fields of 12 characters containing real numbers
- 7 fields of 12 characters containing integer numbers.

Each time history record that follows the header record contains only real numbers, in fields of 12 characters. The fields contain data as specified in Figure I-15. The definitions of the tape data variables along with their corresponding scale factors and coordinates can be found in Appendix P.

HEADER

CHARACTER	CONTEST
1-120	Title
121-132	T - Time of Flight
133-144	RMC(1) - Final Missile Position - Downrange (ft)
145-156	RMC(2) - Final Missile Position - Cross Range
157-168	RMC(3) - Final Missile Position - Altitude
169-180	VM(1) - Final Missile Velocity - Downrange (ft/sec)
181-192	VM(2) - Final Missile Velocity - Cross Range
193-204	VM(3) - Final Missile Velocity - Altitude
205-216	SIM - Final Missile Azimuth Angle (rad)
217-228	THM - Final Missile Elevation Angle
229-240	FIM - Final Missile Roll Angle
241-252	RTC(1) - Final Target Position - Downrange (ft)
253-264	RTC(2) - Final Target Position - Cross Range
265-276	RTC(3) - Final Target Position - Altitude
277-288	VTC(1) - Final Target Velocity - Downrange (ft/sec)
289-300	VTC(2) - Final Target Velocity - Cross Range
301-312	VTC(3) - Final Target Velocity - Altitude

Figure I-15. Digital tape format.

HEADER

CHARACTER	CONTENT
313-324	KTGTSR - Initial Target Slant Range (ft)
325-336	KTGTAL - Initial Target Altitude (ft)
337-348	KTGTSP - Initial Target Speed (ft/sec)
349-360	KTGTAS - Target Aspect (rad)
361-372	KTGTMG - Target Maneuver (g-s)
373-384	KTGTMG - Target Maneuver (g-s)
385-396	KMSLAL - Initial Missile Altitude (ft)
397-408	KMSLSP - Initial Missile Speed (ft/sec)
409-420	KMISSD - Final Miss Distance (ft)
421-432	KMISSV(1) - Miss Component X (ft)
433-444	KMISSV(2) - Miss Component Y (ft)
445-456	KMISSV(3) - Miss Component Z (ft)
457-468	KWSCPW - Weapon Sys Control Panel Word (Integer Code)
469-480	KRUNNO - Run Number
481-492	KTGMD - Target Model
493-504	KTGTSC - Scenario Type
505-516	KTGTSN - Scenario Number
517-528	KTERMC - Termination Code
529-540	KTGTMS - Target Maneuver Sign

Figure I-15.

Tri-FAST DIGITAL TAPE FORMAT

TIME HISTORY DATA

The following data are described in Appendix P. The data, named below, appear in 12 character fields in the order specified:

1. TIME (of flight)
2. NM(1)
3. NM(2)
4. NM(3)
5. VM(1)
6. VM(2)
7. VM(3)
8. RMC(1)
9. RMC(2)
10. RMC(3)
11. WM(1)
12. WM(2)
13. WM(3)
14. SIM
15. THM
16. FIM
17. DL2
18. DL3
19. AL2
20. AL3
21. MVM
22. DLI2
23. DLI3
24. THHC
25. SIHC
26. VG2C
27. VG3C
28. LTMDC
29. [illegible]MC
30. [illegible] (software)
31. SIH (software)
32. VG2 (software)
33. VG3 (software)
34. VTC(1)
35. V[illegible]2)
36. [illegible]3)
37. [illegible]C(1)
38. RTC(2)
39. RTC(3)
40. RAGGS
41. RAGES
42. RPN2
43. RPN3
44. RRG2
45. RRG3
46. VAB2
47. VAB3
48. LAC2 (lambda comp)
49. LAC3 (lambda comp)
50. SITDC
51. TDSRL
52. HBWRL
53. BSWTC
54. RCSC
55. ASP2C
56. ASP3C
57. DOPPC
58. TGDATC(1) (target RF)
59. (2)
60. (3)
61. (4)
62. (5)
63. (6)
64. (7)
65. (8)
66. (9)
67. (10)
68. (11)
69. (12)
70. (13)
71. (14)
72. (15)
73. (16)
74. NA(1) (longitudinal acceleration)
75. LTM (software)
76. LTMD (software)

Figure I-15. (Concluded)

The header record contains the run description, that is, the run number, date, the WSCP word, KWSCPW, which contains the status of the WSCP for this run, the reason for termination of the simulation, KTERMC, etc. KWSCPW is an integer word made up of the switch settings from the upper two rows of the WSCP. The data word bit positions (0-23) corresponding to each switch are indicated in Figure I-16. Bit 0 is the last significant bit, and a bit set indicates that the corresponding switch is on. The reason for simulation termination and the associated integer code are listed below.

- Range - 1
- Passed point of closest approach - 2
- No angle track at launch - 3
- Datacraft manually stopped - 4
- TARFS angle or angle rate limit - 5
- Array - 6
- 6600 stopped simulation - 7
- Minicomputer abort - 8
- Datacraft software (time out, arithmetic overflow, etc.) - 9
- Harris/6 or AP120B - 10
- LSI-11 status error - 11.

The header record also contains the end of run data: miss distance, final values of selected variables, initial conditions for target and missile, etc.

The time history data file literally contains the time histories of selected variables at recording intervals of either 10 msec or 50 msec. A few runs were made at 20 msec instead of 10 msec due to default data storage limitations.

There arises the question of how old the data are with respect to when they were recorded on tape. The following lists the source of a particular set of variables and the variables' relative age with respect to the value of TFLT in the tape record in which the variables are located.

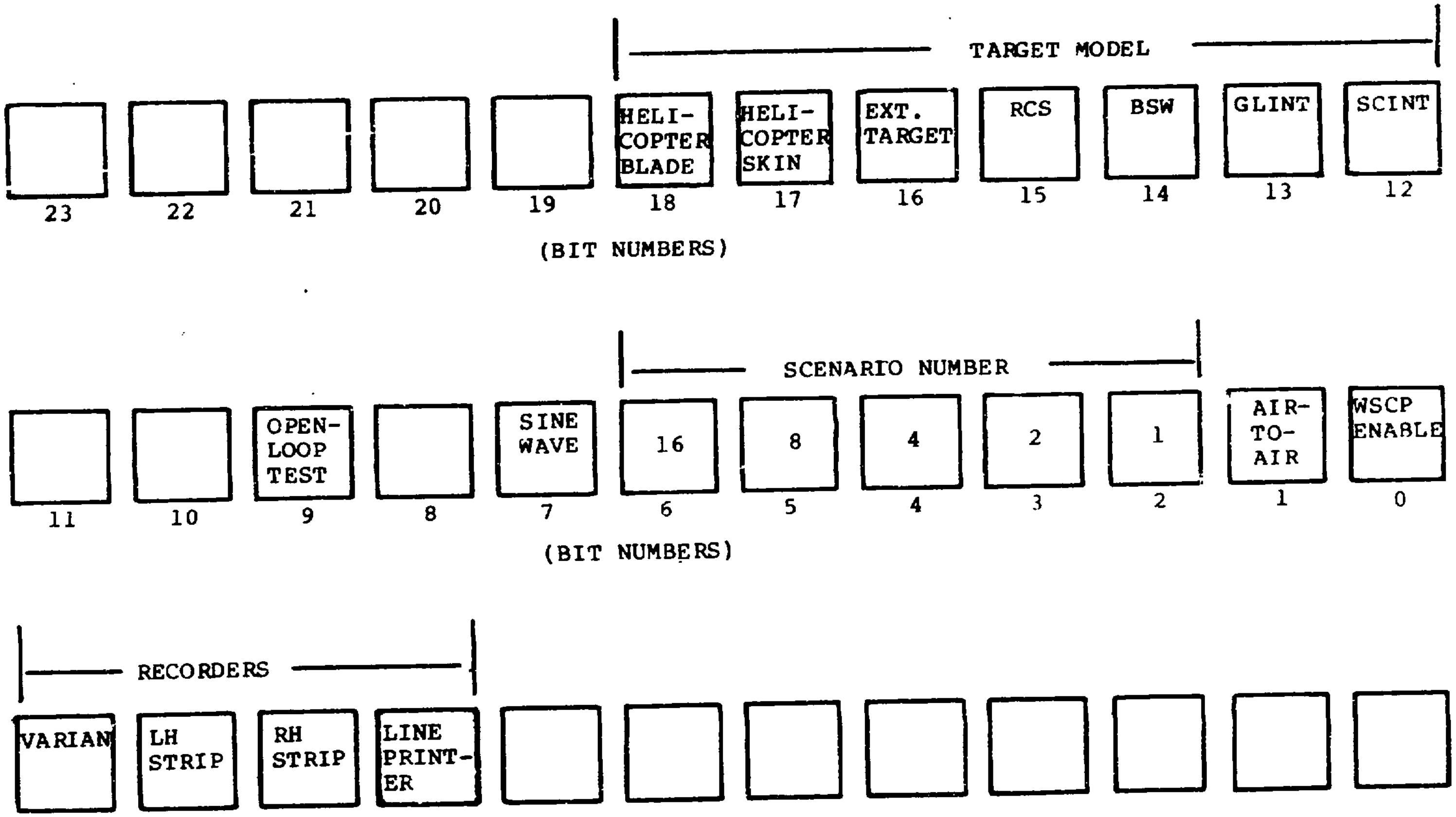
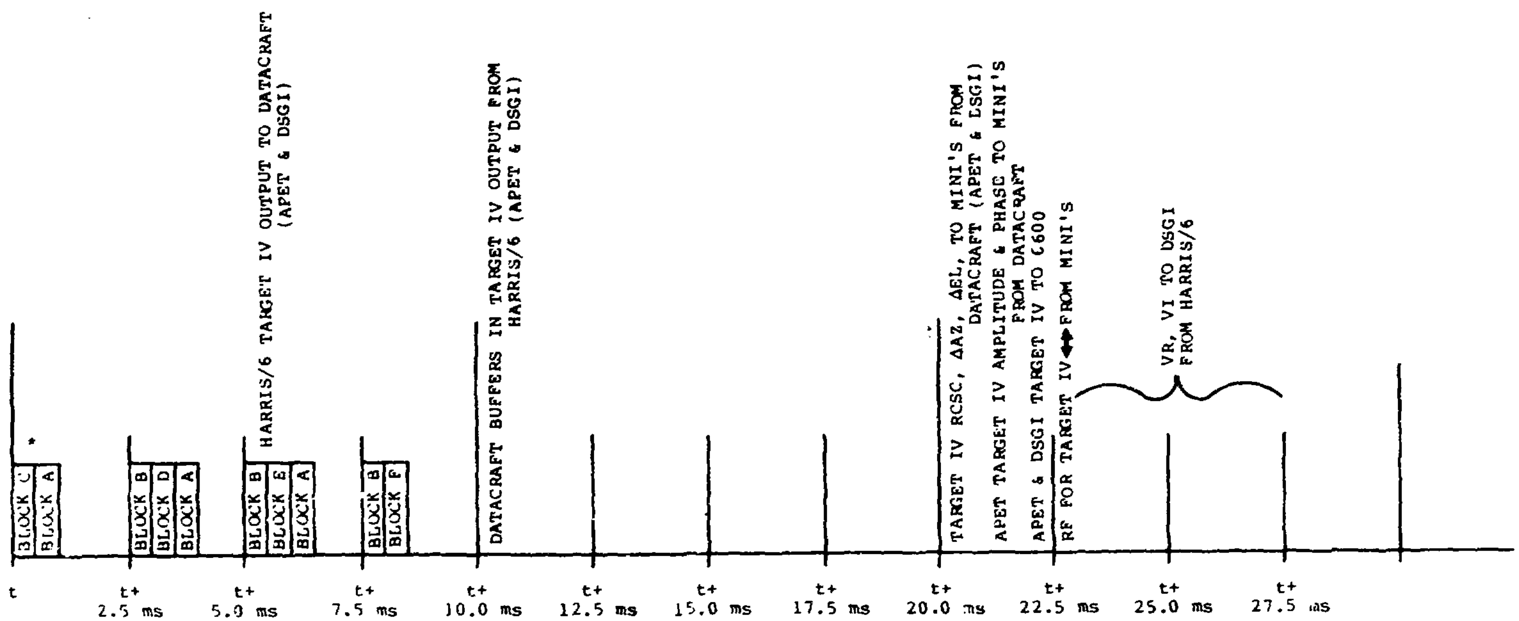


Figure I-16. WSCP switch assignments.

VARIABLE SOURCE	RELATIVE AGE wrt TFLT (10 msec Update Rate)
6600 Integrated Variables (States)	0.0 msec old
6600 Intermediate Variables	3.33 msec old
Analog Variables	10.0 msec old
Non-RF Direct Cell C Data	10.0 msec old
Direct Cell C Data (Targets I, II, III, V)	10.0 msec old from when Datacraft output to Mini's
	7.5 msec old from when Mini's commanded RF hardware
Direct Cell C Data (Target IV-APET)	25.0 msec old from when Datacraft receives input from Harris/6
	7.5 msec old from when Mini's commanded RF hardware (RCSC, AZ, EL, VR, VI, see Appendix P, Figure P-2)
Direct Cell C Data (Target IV - DSGI)	25.0 msec old from when Datacraft receives input from Harris/6
	7.5 msec old from when Mini's commanded hardware (RCSC, AZ, EL, see Appendix P, Figure P-2)
	2.5 - 7.5 msec old from when Harris/6 commanded DSGI hardware (VR, VI, see Appendix P, Figure P-2)

The specific variables and their scaling are discussed in Subsection 4.H. As shown above, not only must one know



*Block sequences are repeated every 10 ms.

Figure I-17. Timing diagram for Datacraft I/O processing.

when the RF variables were generated but also when they were used. To aid in this explanation of the recording age of the variables, a timing diagram was developed as shown in Figure I-17. The diagram is based on the I/O processing of the RFSS Datacraft computer. The corresponding block codes are explained below.

BLOCK A (DATACRAFT)

1. Closing velocity first order hold for RF mini and LSI-11
2. Range first order hold for RF mini and LSI-11
3. Target/missile position first order hold for Harris/6
4. Doppler/range attenuation update using first order hold values
5. Buffer hardware seeker output data

BLOCK B

1. Datacraft output to Harris/6
2. Datacraft output to RF mini
3. Skr hardware I/O w/Datacraft

BLOCK C

1. Harris/6 input from Datacraft
2. 6600 I/O w/Datacraft and analogs
3. Target mini input from Datacraft
4. RF mini input from Datacraft
5. Control mini input from Datacraft
6. Hardware seeker I/O w/Datacraft
7. TARFS input from Datacraft
8. Stripchart input from Datacraft
9. Tektronix 613 input from Datacraft
10. Datacraft buffers in 6600 input from Direct Cell and calculates LOS acceleration

BLOCK D (DATACRAFT)

1. Target velocity, Euler rates calculated
2. All integrations performed
3. Inertial to body transformations
4. TARFS angles

BLOCK E (DATACRAFT)

1. Relative/array geometry
2. Inertial to line-of-sight and target to line-of-sight transformations

BLOCK F (DATACRAFT)

1. Target Models I, II, III, IV, and V parameters output
2. End condition checks
3. Update and buffer all outputs

D. ACSL POSTPROCESSOR PLOTS OF HARDWARE-IN-THE-LOOP SIMULATION DATA

The Tri-FAST hardware seeker was tested with a closed-loop missile simulation in a variety of scenarios with targets of varying electromagnetic complexity. According to the variation in miss distance and the complexity of the target, 10 to 50 Monte Carlo runs were made for each scenario.

During the intercept sequences, a data file was built up on the CDC 6600 disk for each flight and periodically dumped to tape when about 2 million words had been recorded. Plots of selected data from the magnetic tapes were obtained using the ACSL Postprocessor.

Since several flights in the recorded data were aborts, the actual Monte Carlo sequence of flights to be processed must be specified for each Postprocessor action. Flights were described by a unique run number and the runs to be analyzed were specified to the Postprocessor.

PROCESS 356, 360

PROCESS 364

will accumulate runs 356 through 360 and run 364 into an ensemble set to be postprocessed. The FAST customers selected the runs to be postprocessed. Figure I-18 contains the variables plotted. The guidance and acceleration variables actually plotted are the time average $\bar{x}_i$ and the expected band $\bar{x}_i-2.5\sigma_i$ and $\bar{x}_i+2.5\sigma_i$ where σ_i is the standard deviation. The ensemble or time average $\bar{x}_i$ is calculated from

$$\bar{x}_i = \frac{1}{N} \sum_{j=1}^{N} x_{ij}$$

where $\bar{x}_{ij}$ is the ith data point in the jth intercept and N is the number of intercepts designated (by the PROCESS command) for ensemble averaging. The standard deviation is calculated from

$$\sigma_i^2 = \frac{1}{N} \sum_{j=1}^{N} x_{ij}^2 - \bar{x}_i^2 \quad .$$

Figure I-19 shows the coordinate system for miss distance in a plane normal to the missile-from-target relative velocity vector.

Figure I-20 is a list of every PROCESS command used. Notes have been added as necessary if the data produced abnormal results or special scaling had to be used. Volume V contains the plots and the programming procedures used to generate them.

E. DESCRIPTION OF THE COORDINATE SYSTEMS USED BY THE RFSS DATACRAFT COMPUTER

(1) <u>INERTIAL SYSTEM</u> (X_I,Y_I,Z_I). The origin of this system is at ground level for the surface-to-air missile and at sea level for the air-to-air missile and is directly below the missile launch point. The X_I axis is in the ground plane and is positive downrange. The positive Z_I axis is down (positive towards the center of earth). Y_I completes a right hand (RH) system (Figure I-21).

Transformations between this and other systems are defined by the Euler angles ψ, θ, ϕ with

ORDINATE	ABSCISSA	COORDINATE SYSTEM	ORDINATE UNITS
VG2CZA - U/D Guidance Command (El Gyro Torquer)	T-time (sec)	Missile Body	Deg/sec
VG3CZA - R/L Guidance Command (Az Gyro Torquer)	T-time (sec)	Missile Body	Deg/sec
NM2ZA - R/L Acceleration	T-time (sec)	Missile Body	G's
NM3ZA - U/D Acceleration	T-time (sec)	Missile Body	G's
RMTT3F - Miss Distance along Z-Axis	RMTT1F - Miss Distance along X-Axis	Target	Ft
RMTT2F - Miss Distance along Y-Axis	RMTT1F - Miss Distance along X-Axis	Target	Ft
RMTV3F - Miss Distance along Z-Axis	RMTV2F - Miss Distance along Y-Axis	YZ Plane normal to relative velocity vector (velocity of missile - velocity of target in the earth frame), the Y-axis is parallel to the earth and the Z-axis points downward (See Figure I-19)	Ft

Figure I-18. Monte Carlo Jata variables plotted.

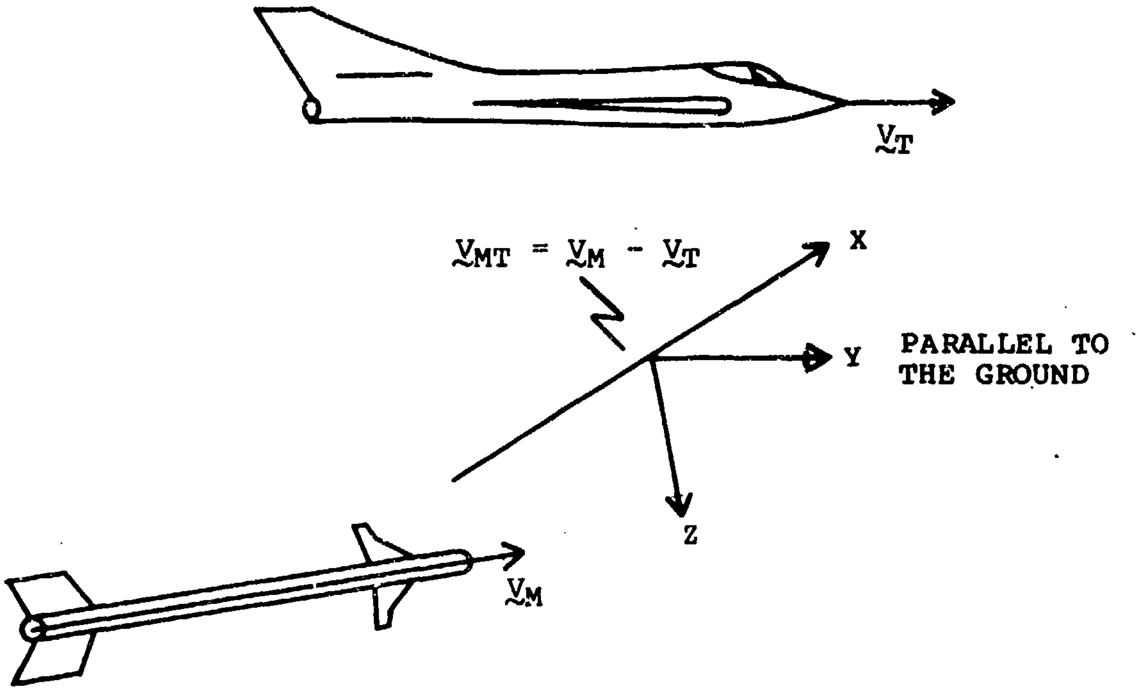


Figure I-19. Illustration of the coordinate system for miss distance in a plane (Y-Z) normal to the missile-from-target relative velocity vector.

ψ a RH rotation about Z_I taking $X \rightarrow X'$; $Y \rightarrow Y'$

θ a RH rotation about Y' taking $X' \rightarrow X''$; $Z_I \rightarrow Z'$

ϕ a RH rotation about X'' taking $Y' \rightarrow Y''$; $Z' \rightarrow Z''$.

(2) LAB COORDINATE SYSTEM (X_L, Y_L, Z_L). Flight table (TARFS-1) angles and target position on the array are calculated in this system (Figure I-22). The origin of the system is at the intersection of the flight table gimbal axes.

The positive X_L axis points toward the center of the array; Z_L is positive down and Y_L completes a RH system. This is an inertial system which translates with the missile. The missile-target relative geometry is evaluated in this system and array angles are specified in the system as shown in Figure I-23. The initial orientation of the Lab coordinates with respect to the Inertial coordinates is specified by BSIRFZ and BTHRFZ in the Datacraft nine printer listing (Figure I-5).

```
REQUEST(FILE,MT)          S11175
"LABEL"="AA3-AD-ROLL-I/RUNS 277, 292"      Runs 274-276  1
SPARE PROCESS,  277
SPARE PROCESS,  283 ,  285
SPARE PROCESS,   289,  292
SPARE GO $ "CMD"=9
"LABEL"="AA7-AD-NOROLL-I/RUNS 298,312"
SPARE PROCESS,   298
SPARE PROCESS,   301,  306
SPARE PROCESS,   308,  309
SPARE PROCESS,   312
SPARE GO $ "CMD"=9
"LABEL"="SA9-AD-NOROLL-I/RUNS 314, 335"
SPARE PROCESS,   314,  315                   314  1
SPARE PROCESS,   320
SPARE PROCESS,   326
SPARE PROCESS,   329,  330
SPARE PROCESS,   332,  335
SPARE GO $ "CMD"=9
"LABEL"="AA3-AD-NOROLL-I/RUNS 376, 392"
SPARE PROCESS,  376,  380                    376  1
SPARE PROCESS,   382,  383
SPARE PROCESS,   386,  387
SPARE PROCESS,   390                         390 & 392  3
SPARE PROCESS,   392
SPARE GO $ "CMD"=9
"LABEL"="AA2-AD-NOROLL-I/RUNS 394, 405"
SPARE PROCESS,   394,  395
SPARE PROCESS,   397
SPARE PROCESS,   400,  401
SPARE PROCESS,   403
SPARE PROCESS,   405
SPARE GO $ "CMD"=9
```

1 Run not on tape

2 Two runs (one good, one bad) given same run number

3 Miss distance different than shown on run log

4 Time data recorded incorrectly on tape

5 Bad data

Runs 277-493 have scale factors for the Y and Z miss distance components that made them four times larger than they actually are.

Figure I-20. Run numbers used in Postprocessor plots.

```
"LABEL"="SA2-AD-NOROLL-I/RUNS 430, 443"
SPARE PROCESS,   430,  434
SPARE PROCESS,   436,  437
SPARE PROCESS,   439
SPARE PROCESS,   442,  443
SPARE GO $ "CMD"=9
"LABEL"="AA3-AH-NOROLL-I/RUNS 454, 466"
SPARE PROCESS,   454
SPARE PROCESS,   456
SPARE PROCESS,   458
SPARE PROCESS,   460
SPARE PROCESS,  462,  463
SPARE PROCESS,   465,  466
SPARE GO $ "CMD"=9
"LABEL"="AA3-AD-NOROLL-I/RUNS 477, 493"
SPARE PROCESS,   477                              477  1
SPARE COMPRESS,  480
SPARE COMPRESS,  483
SPARE PROCESS,   485,  486
SPARE PROCESS,   489,  493                        493 not plotted
SPARE GO $ "CMD"=9
REQUEST(FILE,MT)        S06634
"LABEL"="AA7-AH-ROLL-I/RUNS 534, 558"
SPARE PROCESS,   534,  535
SPARE PROCESS,   537
SPARE PROCESS,   541,  542
SPARE COMPRESS,  547
SPARE COMPRESS,  548
SPARE COMPRESS,  550
SPARE PROCESS,   554
SPARE PROCESS,   558
SPARE GO $ "CMD"=9
"LABEL"="AA4-AD-ROLL-I/RUNS 583, 599"
SPARE PROCESS,   583,  584
SPARE COMPRESS,  585
SPARE COMPRESS,  586
SPARE PROCESS,   587,  588
SPARE PROCESS,   590
SPARE PROCESS,  593
SPARE PROCESS,  598,  599
SPARE GO $ "CMD"=9
```

Figure I-20.

```
"LABEL"="SA14-AD-NOROLL-I/RUNS 659, 681"
SPARE PROCESS,   659
SPARE PROCESS,   663,  665
SPARE PROCESS,   670
SPARE PROCESS,   674
SPARE PROCESS,   676,  677
SPARE PROCESS,   680,  681
SPARE GO $ "CMD"=9
ATTACH,FILE,MAGTAPE,ID=DUXXXM.
"LABEL"="SA14-AD-ROLL-I/RUNS 682, 719"
SPARE PROCESS,   682
SPARE PROCESS,   689,  691                          697-719  1
SPARE PROCESS,   697
SPARE PROCESS,   704
SPARE PROCESS,   709
SPARE PROCESS,   712
SPARE PROCESS,   714
SPARE PROCESS,   719
SPARE GO $ "CMD"=9
REQUEST(FILE,MT)          S02886
"LABEL"="SA14-CD-NOROLL-I/RUNS 723, 734"
SPARE PROCESS,  723,  727
SPARE PROCESS,   729,  730
SPARE PROCESS,   732,  734
SPARE GO $ "CMD"=9
"LABEL"="SA14-CD-ROLL-I/RUNS 750, 768"
SPARE PROCESS,   750,  753
SPARE PROCESS,   755,  756
SPARE PROCESS,   758,  759
SPARE PROCESS,   764,  765
SPARE PROCESS,   767,  768
SPARE GO $ "CMD"=9
"LABEL"="SA10-AD-ROLL-I/RUNS 772, 794"
SPARE PROCESS,   772
SPARE PROCESS,   777,  778
SPARE PROCESS,   780,  782
SPARE COMPRESS,  785
SPARE COMPRESS,  786
SPARE PROCESS,   789
SPARE PROCESS,   794
SPARE GO $ "CMD"=9
"LABEL"="SA8-AD-NOROLL-I/RUNS 821, 833"
SPARE PROCESS,   821,  822
SPARE PROCESS,   824,  829
SPARE PROCESS,   832       833  20 msec run - not processed
SPARE GO $ "CMD"=9
```

Figure I-20.

```
"LABEL"="SA8-AD-ROLL-I/RUNS 835, 847"
SPARE PROCESS,   835
SPARE PROCESS,   837,  840
SPARE PROCESS,   842,  847
SPARE GO $ "CMD"=9
"LABEL"="AA3-AD-ROLL-II/RUNS 857, 869"
SPARE PROCESS,   857,  858                        858  3
SPARE PROCESS,   860,  863
SPARE COMPRESS,  864
SPARE COMPRESS,  865
SPARE PROCESS,   867
SPARE PROCESS,   869
SPARE GO $ "CMD"=9
"LABEL"="AA7-AD-NOROLL-II/RUNS 889, 907"
SPARE PROCESS,   889
SPARE PROCESS,   892
SPARE PROCESS,   895
SPARE PROCESS,   897,  900
SPARE COMPRESS,  904
SPARE COMPRESS,  905
SPARE COMPRESS,  907
SPARE GO $ "CMD"=9
REQUEST(FILE,MT)        S12185/PLEASE DO NOT USE MT33 OR MT35
COPYBF(FILE,TEMP)
"LABEL"="SA14-AD-NOROLL-II/RUNS 908, 920"
SPARE PROCESS,   908,  910                        909  1
SPARE PROCESS,   912
SPARE PROCESS,  914
SPARE COMPRESS,  917
SPARE COMPRESS,  918
SPARE PROCESS,   919,  920
SPARE GO $ "CMD"=9
"LABEL"="SA10-AD-NOROLL-II/RUNS 939, 979"
SPARE PROCESS,   939
SPARE PROCESS,   943
SPARE PROCESS,   950
SPARE PROCESS,   952
SPARE PROCESS,   955,  956
SPARE PROCESS,   965
SPARE PROCESS,   968
SPARE COMPRESS,  974                              974  1
SPARE COMPRESS,  979
SPARE GO $ "CMD"=9
```

Figure I-20.

```
"LABEL"="AA3-AD-NOROLL-II/RUNS 982, 994"
SPARE PROCESS,   982,  986
SPARE COMPRESS,  988
SPARE COMPRESS,  990
SPARE PROCESS,   991
SPARE PROCESS,   993,  994
SPARE GO $ "CMD"=9
"LABEL"="[illegible]-AD-NOROLL-II/RUNS 997, 1010"
SPARE PROCESS,   997
SPARE PROCESS,   999, 1001
SPARE COMPRESS, 1003
SPARE COMPRESS, 1004
SPARE PROCESS,  1005
SPARE PROCESS,  1008, 1010
SPARE GO $ "CMD"=9
"LABEL"="SA2-AD-NOROLL-II/RUNS 1012, 1030"
SPARE PROCESS,  1012
SPARE PROCESS,  1019
SPARE PROCESS,  1021
SPARE PROCESS,  1023, 1024
SPARE COMPRESS, 1025
SPARE COMPRESS, 1027                                 1027-1043  1
SPARE PROCESS,  1028, 1030
SPARE GO $ "CMD"=9
"LABEL"="SA2-AD-ROLL-I/RUNS 1031, 1043"
SPARE PROCESS,  1031, 1032
SPARE PROCESS,  1034
SPARE PROCESS,  1037
SPARE COMPRESS, 1038
SPARE COMPRESS, 1039
SPARE PROCESS,  1040, 1043
SPARE GO $ "CMD"=9
REQUEST(FILE,MT)          S05728
"LABEL"="SA5-AD-NOROLL-I/RUNS 1049, 1065"
SPARE PROCESS,  1049, 1051
SPARE PROCESS,  1054
SPARE COMPRESS, 1058
SPARE COMPRESS, 1060
SPARE PROCESS,  1061, 1063
SPARE PROCESS,  1065
SPARE GO $ "CMD"=9
```

Figure I-20.

```
"LABEL"="SAS-AD-NOROLL-II/RUNS 1066, 1076"
SPARE PROCESS,  1066, 1070
SPARE COMPRESS, 1071
SPARE COMPRESS, 1072
SPARE PROCESS,  1073
SPARE PROCESS,  1075, 1076
SPARE GO $ "CMD"=9
"LABEL"="AA2-AD-ROLL-I/RUNS 1105, 1126"
SPARE PROCESS,  1105, 1106
SPARE PROCESS,  1113
SPARE PROCESS,  1115
SPARE PROCESS,  1117
SPARE COMPRESS, 1121
SPARE COMPRESS, 1122
SPARE PROCESS,  1123, 1124
SPARE PROCESS,  1126
SPARE GO $ "CMD"=9
"LABEL"="AA2-AD-NOROLL-I/RUNS 1127, 1146"
SPARE PROCESS,  1127
SPARE PROCESS,  1131, 1133
SPARE PROCESS,  1137, 1139
SPARE PROCESS,  1141
SPARE COMPRESS, 1143
SPARE COMPRESS, 1146
SPARE GO $ "CMD"=9
"LABEL"="AA3-AH-NOROLL-I/RUNS 1155, 1164"
SPARE PROCESS,  1155, 1164
SPARE GO $ "CMD"=9
"LABEL"="AA3-AD-NOROLL-III/RUNS 1170, 1195"
SPARE PROCESS,  1170
SPARE PROCESS,  1172, 1177                       1172   3
SPARE PROCESS,  1179
SPARE PROCESS,  1181
SPARE PROCESS,  1183, 1186
SPARE PROCESS,  1188                             1188-1195   4
SPARE COMPRESS, 1190
SPARE COMPRESS, 1191
SPARE PROCESS,  1192, 1195
SPARE GO $ "CMD"=9
```

Figure I-20.

```
REQUEST(FILE,MT)        S02061/PLEASE DO NOT USE MT33 OR MT35
COPYBF(FILE,TEMP)
"LABEL"="AA3-AH-NOROLL-III/RUNS 1199, 1259"
$PARE PROCESS,  1199, 1200
$PARE PROCESS,  1202, 1204
$PARE PROCESS,  1206, 1207
$PARE PROCESS,  1209, 1211
$PARE PROCESS,  1213, 1215
$PARE PROCESS,  1217, 1220                        1219  1
$PARE PROCESS,  1222
$PARE COMPRESS, 1223
$PARE COMPRESS, 1224
$PARE PROCESS,  1225, 1229
$PARE PROCESS,  1231, 1232
$PARE PROCESS,  1234, 1243
$PARE PROCESS,  1246, 1250
$PARE PROCESS,  1252, 1259
$PARE GO $ "CMD"=9
"LABEL"="AA7-AD-NOROLL-III/RUNS 1279, 1343"
$PARE PROCESS,  1279, 1281
$PARE PROCESS,  1284
$PARE PROCESS,  1286, 1288
$PARE PROCESS,  1290
$PARE PROCESS,  1292, 1300
$PARE PROCESS,  1302
$PARE PROCESS,  1304
$PARE COMPRESS, 1305
$PARE COMPRESS, 1306
$PARE PROCESS,  1309
$PARE PROCESS,  1311, 1321
$PARE PROCESS,  1323, 1324
$PARE COMPRESS, 1325
$PARE COMPRESS, 1327
$PARE PROCESS,  1328, 1335
$PARE PROCESS,  1339, 1343                        1343  3
$PARE GO $ "CMD"=9
"LABEL"="AA7-AD-NOROLL-I/RUNS 1345, 1355"
$PARE PROCESS,  1345, 1352
$PARE COMPRESS, 1355                              1354  2
$PARE GO $ "CMD"=9
```

Figure I-20.

```
REQUEST(FILE,MT)         S08082/PLEASE DO NOT USE MT33 OR MT35
COPYBF(FILE,TEMP)
"LABEL"="SA14-AD-NOROLL-I/RUNS 1356, 1369"
SPARE PROCESS,  1356, 1357
SPARE PROCESS,  1359
SPARE PROCESS,  1363, 1364
SPARE COMPRESS, 1365
SPARE COMPRESS, 1366
SPARE PROCESS,  1367, 1369
SPARE GO $ "CMD"=9
"LABEL"="SA14-AD-NOROLL-III/RUNS 1370, 1438"
SPARE PROCESS,  1370, 1371
SPARE PROCESS,  1374, 1375
SPARE PROCESS,  1378, 1379                          1377  2
SPARE PROCESS,  1381, 1385
SPARE PROCESS,  1387, 1389                          1389  3
SPARE PROCESS,  1391
SPARE COMPRESS, 1392
SPARE COMPRESS, 1393
SPARE COMPRESS, 1394
SPARE PROCESS,  1395, 1396
SPARE PROCESS,  1400, 1401
SPARE PROCESS,  1404, 1408
SPARE PROCESS,  1410, 1412
SPARE PROCESS,  1415, 1425
SPARE PROCESS,  1428, 1429
SPARE PROCESS,  1431, 1432
SPARE PROCESS,  1434
SPARE PROCESS,  1436, 1438                          1436  1
SPARE GO $ "CMD"=9
REQUEST(FILE,MT)         S08421
"LABEL"="SA2-AD-NOROLL-III/RUNS 1443, 1485"
SPARE PROCESS,  1443
SPARE PROCESS,  1446, 1452
SPARE PROCESS,  1455, 1460
SPARE COMPRESS, 1461
SPARE COMPRESS, 1463
SPARE PROCESS,  1464, 1466
SPARE PROCESS,  1468, 1473
SPARE PROCESS,  1476
SPARE PROCESS,  1478, 1480
SPARE PROCESS,  1485
SPARE GO $ "CMD"=9
```

Figure I-20.

```
"LABEL"="AA3-AD-NOROLL-I/RUNS 1492, 1503"
SPARE PROCESS,  1492, 1495                          1497  5
SPARE COMPRESS, 1498                                1498  3
SPARE COMPRESS, 1499                                1499  1
SPARE COMPRESS, 1500
SPARE PROCESS,  1501, 1503
SPARE GO $ "CMD"=9
REQUEST(FILE,MT)        S015/91PLEASE DO NOT USE MT33 OR MT35
"LABEL"="SA2-AD-NOROLL-I/RUNS 1504, 1527"
SPARE PROCESS,  1504, 1506
SPARE PROCESS,  1509, 1510
SPARE PROCESS,  1518
SPARE PROCESS,  1520
SPARE COMPRESS, 1522
SPARE COMPRESS, 1524
SPARE PROCESS,  1527
SPARE GO $ "CMD"=9
"LABEL"="SA10-AD-NOROLL-III/RUNS 1534, 1630"
SPARE PROCESS,  1534
SPARE PROCESS,  1536
SPARE PROCESS,  1538
SPARE PROCESS,  1540
SPARE PROCESS,  1542, 1544
SPARE PROCESS,  1547, 1548
SPARE PROCESS,  1551, 1553
SPARE PROCESS,  1556
SPARE PROCESS,  1566
SPARE PROCESS,  1568
SPARE PROCESS,  1571, 1573
SPARE COMPRESS, 1577
SPARE COMPRESS, 1578
SPARE PROCESS,  1580
SPARE PROCESS,  1582, 1583
SPARE PROCESS,  1585, 1587
SPARE PROCESS,  1589, 1591
SPARE PROCESS,  1594
SPARE PROCESS,  1597, 1601
SPARE PROCESS,  1603, 1606
SPARE PROCESS,  1608, 1609
SPARE PROCESS,  1613
SPARE PROCESS,  1616, 1617
SPARE PROCESS,  1619, 1620
SPARE PROCESS,  1624, 1626
SPARE PROCESS,  1630
SPARE GO $ "CMD"=9
```

Figure I-20.

```
REQUEST(FILE,MT)         S08551
"LABEL"="SA10-AD-NOROLL-I/RUNS 1631, 1707"
SPARE PROCESS,  1631, 1633
SPARE PROCESS,  1635
SPARE PROCESS,  1641
SPARE PROCESS,  1653, 1654
SPARE COMPRESS, 1655
SPARE COMPRESS, 1656
SPARE PROCESS,  1659
SPARE GO $ "CMD"=9
"LABEL"="AA2-AD-NOROLL-III/RUNS 1685, 1707"
SPARE PROCESS,  1685
SPARE PROCESS,  1690, 1691
SPARE PROCESS,  1695, 1699
SPARE PROCESS,  1707
SPARE GO $ "CMD"=9
"LABEL"="AA2-AD-NOROLL-III/RUNS 1730, 1815"
SPARE PROCESS,  1730
SPARE PROCESS,  1734
SPARE PROCESS,  1736, 1737
SPARE PROCESS,  1746, 1748
SPARE PROCESS,  1750, 1756
SPARE PROCESS,  1758, 1762
SPARE COMPRESS, 1763
SPARE COMPRESS, 1765
SPARE PROCESS,  1767, 1769
SPARE PROCESS,  1771, 1772
SPARE PROCESS,  1774
SPARE PROCESS,  1777
SPARE PROCESS,  1779, 1784
SPARE PROCESS,  1786, 1788
SPARE PROCESS,  1795, 1797
SPARE PROCESS,  1800, 1802
SPARE PROCESS,  1804
SPARE PROCESS,  1806
SPARE PROCESS,  1808
SPARE PROCESS,  1810
SPARE PROCESS,  1813, 1815
SPARE GO $ "CMD"=9
```

Figure I-20.

```
REQUEST(FILE,MT)          S07051
"LABEL"="SA5-AD-NOROLL-III   NS 1822, 1877"
SPARE PROCESS,  1822
SPARE PROCESS,  1824
SPARE PROCESS,  1826, 1827
SPARE PROCESS,  1830
SPARE PROCESS,  1832, 1833
SPARE PROCESS,  1835
SPARE PROCESS,  1840
SPARE PROCESS,  1845, 1848
SPARE COMPRESS, 1850
SPARE COMPRESS, 1851
SPARE PROCESS,  1852
SPARE PROCESS,  1854, 1858
SPARE PROCESS,  1860
SPARE PROCESS,  1863, 1864
SPARE PROCESS,  1866, 1870                            1870   1
SPARE PROCESS,  1877
SPARE GO $ "CMD"=9
REQUEST(FILE,MT)          S08583
"LABEL"="AA4-AD-NOROLL-I/RUNS 1884, 1897"
SPARE PROCESS,  1884
SPARE PROCESS,  1887, 1889
SPARE COMPRESS, 1891
SPARE COMPRESS, 1892
SPARE PROCESS,  1894, 1897
SPARE GO $ "CMD"=9
"LABEL"="AA4-AD-NOROLL-II/RUNS 1898, 1910"
SPARE PROCESS,  1898, 1901
SPARE PROCESS,  1903, 1904
SPARE COMPRESS, 1905
SPARE COMPRESS, 1906
SPARE COMPRESS, 1907
SPARE PROCESS,  1910                                  1910   1
SPARE GO $ "CMD"=9
```

Figure I-20.

```
"LABEL"="AA4-AD-NOROLL-III/RUNS 1911, 1949"
SPARE COMPRESS, 1911
SPARE COMPRESS, 1912
SPARE PROCESS,  1915, 1916
SPARE PROCESS,  1918, 1926
SPARE PROCESS,  1928
SPARE PROCESS,  1930, 1932
SPARE COMPRESS, 1933
SPARE COMPRESS, 1934                                1934  1
SPARE PROCESS,  1935
SPARE PROCESS,  1937, 1938
SPARE PROCESS,  1941, 1947
SPARE PROCESS,  1949
SPARE GO $ "CMD"=9
REQUEST(FILE,MT)         S12343
"LABEL"="SA5-AD-ROLL-I/RUNS 1951, 1974"
SPARE PROCESS,  1951
SPARE PROCESS,  1955
SPARE PROCESS,  1957
SPARE PROCESS,  1959, 1961
SPARE COMPRESS, 1962
SPARE PROCESS,  1966, 1967
SPARE PROCESS,  1974
SPARE GO $ "CMD"=9
"LABEL"="SA5-AD-ROLL-III/RUNS 1975, 1999"
SPARE PROCESS,  1975
SPARE PROCESS,  1977, 1978
SPARE PROCESS,  1981
SPARE PROCESS,  1983
SPARE COMPRESS, 1984
SPARE COMPRESS, 1985
SPARE PROCESS,  1989, 1991
SPARE PROCESS,  1994
SPARE PROCESS,  1999
SPARE GO $ "CMD"=9
"LABEL"="SA8-AH-NOROLL-I/RUNS 2038, 2064"
SPARE PROCESS,  2038
SPARE PROCESS,  2040
SPARE PROCESS,  2042, 2043
SPARE PROCESS,  2046, 2047
SPARE PROCESS,  2049
SPARE PROCESS,  2051
SPARE COMPRESS, 2055                                2064  4
SPARE GO $ "CMD"=9
```

Figure I-20.

```
REQUEST(FILE,MT)      S07576/PLEASE DO NOT USE MT33 OR MT35
"LABEL"="SA8-AH-NOROLL-V/RUNS 2065, 2130"
SPARE PROCESS,  2065
SPARE PROCESS,  2069
SPARE PROCESS,  2071, 2072
SPARE PROCESS,  2076
SPARE PROCESS,  2079
SPARE PROCESS,  2081
SPARE PROCESS,  2083, 2084
SPARE PROCESS,  2086, 2090
SPARE PROCESS,  2094, 2098
SPARE PROCESS,  2102
SPARE PROCESS,  2104                2105, 2107, 2109, 2115  5
SPARE PROCESS,  2116
SPARE PROCESS,  2124
SPARE PROCESS,  2126
SPARE PROCESS,  2130
SPARE GO $ "CMD"=9
"LABEL"="SA14-CH-NOROLL-III/RUNS 2134, 2202"
SPARE PROCESS,  2134
SPARE PROCESS,  2136
SPARE PROCESS,  2138, 2139
SPARE PROCESS,  2144, 2145          2175-2202  1  or  5
SPARE PROCESS,  2147, 2148
SPARE PROCESS,  2154, 2156
SPARE PROCESS,  2161
SPARE PROCESS,  2163, 2167
SPARE COMPRESS, 2168
SPARE GO $ "CMD"=9
"LABEL"="SA14-CH-NOROLL-I/RUNS 2203, 2208"
SPARE PROCESS,  2203, 2205          2206  4
SPARE COMPRESS, 2207                2207, 2208  1
SPARE PROCESS,  2208
SPARE GO $ "CMD"=9
"LABEL"="AA3-AD-NOROLL-I/RUNS 2221, 2231"
SPARE PROCESS,  2221, 2225
SPARE COMPRESS, 2227
SPARE COMPRESS, 2228
SPARE PROCESS,  2229, 2231
SPARE GO $ "CMD"=9
```

Figure I-20.

```
"LABEL"="AA2-AD-NOROLL-I/RUNS 2236, 2258"
$PARE PROCESS,  2236
$PARE PROCESS,  2239
$PARE PROCESS,  2241, 2242
$PARE COMPRESS, 2245
$PARE COMPRESS, 2247
$PARE PROCESS,  2250
$PARE PROCESS,  2252
$PARE PROCESS,  2256
$PARE PROCESS,  2258
$PARE GO $ "CMD"=9
$REQUEST(FILE,MT)           S12230
"LABEL"="AA3-CD-NOROLL-I/RUNS 2279, 2291"
$PARE PROCESS,  2279, 2280
$PARE PROCESS,  2282, 2283
$PARE PROCESS,  2285
$PARE COMPRESS, 2287
$PARE COMPRESS, 2288
$PARE PROCESS,  2289, 2291
$PARE GO $ "CMD"=9
"LABEL"="AA3-CH-NOROLL-III/RUNS 2293, 2384"
$PARE PROCESS,  2293, 2296
$PARE PROCESS,  2301, 2303
$PARE PROCESS,  2308
$PARE PROCESS,  2310, 2312
$PARE PROCESS,  2314
$PARE PROCESS,  2316, 2317
$PARE PROCESS,  2319
$PARE PROCESS,  2321
$PARE PROCESS,  2324, 2325
$PARE PROCESS,  2328, 2329
$PARE COMPRESS, 2330
$PARE COMPRESS, 2332
$PARE PROCESS,  2333
$PARE PROCESS,  2335, 2336
$PARE PROCESS,  2338
$PARE PROCESS,  2346, 2348
$PARE PROCESS,  2353, 2360
$PARE PROCESS,  2363
$PARE PROCESS,  2365, 2366
$PARE PROCESS,  2368, 2369
$PARE PROCESS,  2372, 2374
$PARE PROCESS,  2376
$PARE PROCESS,  2379
$PARE PROCESS,  2382, 2384
$PARE GO $ "CMD"=9
```

Figure I-20.

```
REQUEST(FILE,MT)          S11088
"LABEL"="AA3-CH-NOROLL-I/RUNS 2385, 2401"
SPARE PROCESS,  2385, 2387                        2385  1
SPARE PROCESS,  2389
SPARE PROCESS,  2393, 2394
SPARE COMPRESS, 2395
SPARE COMPRESS, 2396
SPARE PROCESS,  2397
SPARE PROCESS,  2401
SPARE GO $ "CMD"=9
```

Figure I-20. (Concluded)

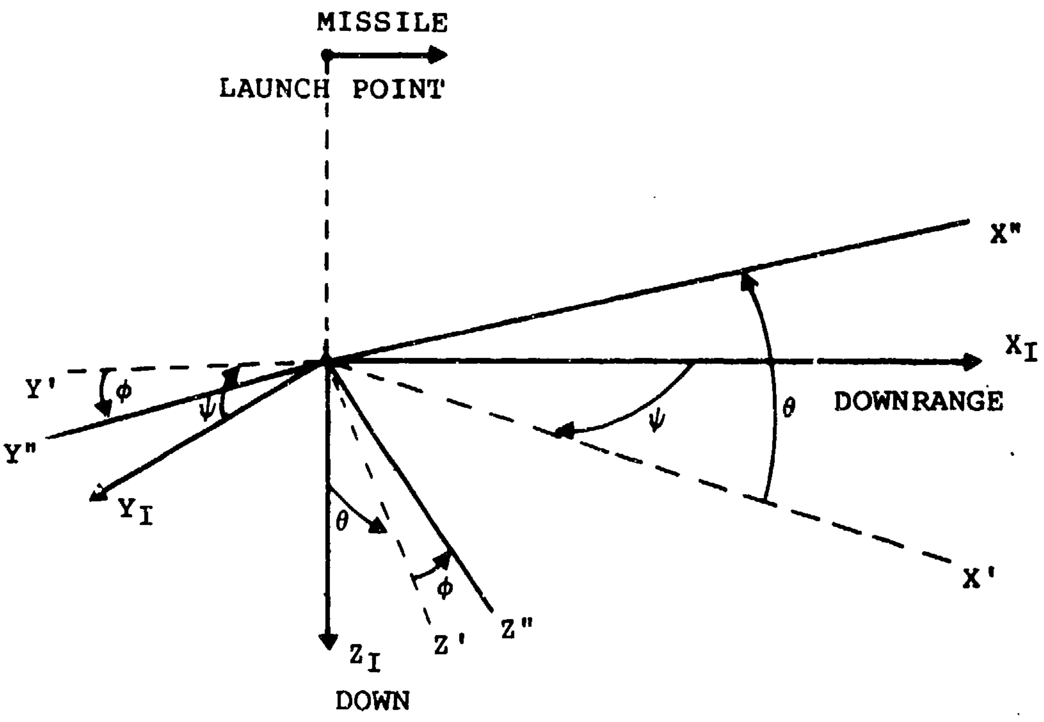


Figure I-21. Euler sequence for Inertial coordinates.

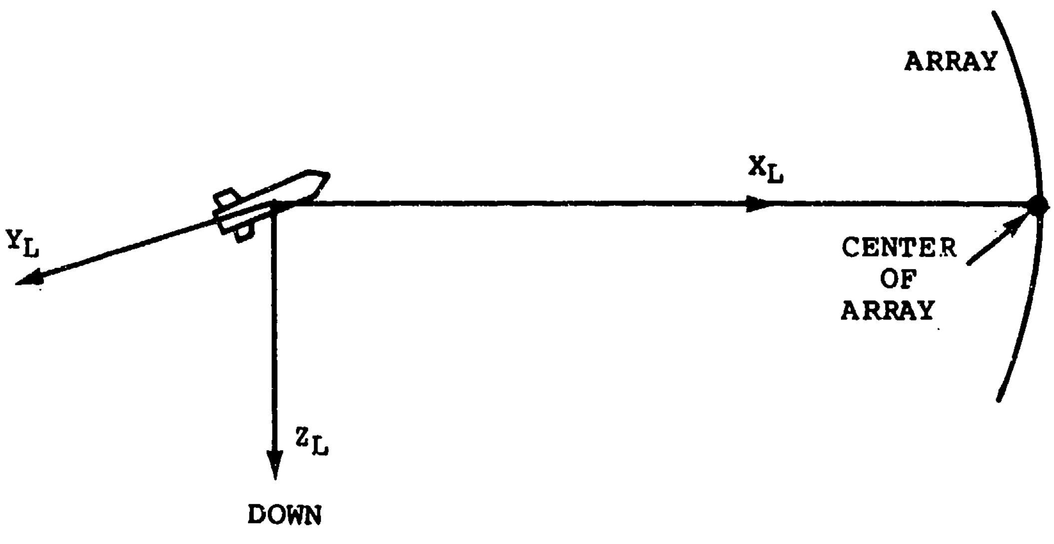


Figure I-22. Lab coordinates description.

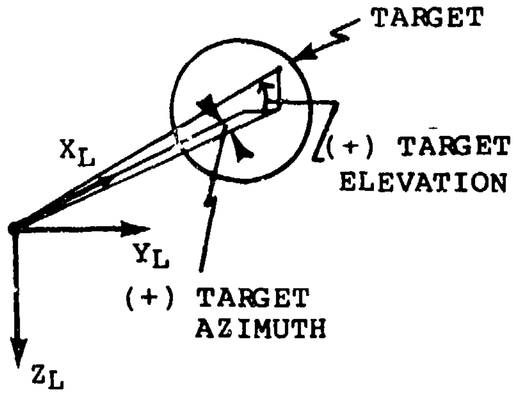


Figure I-23. Array angles definition.

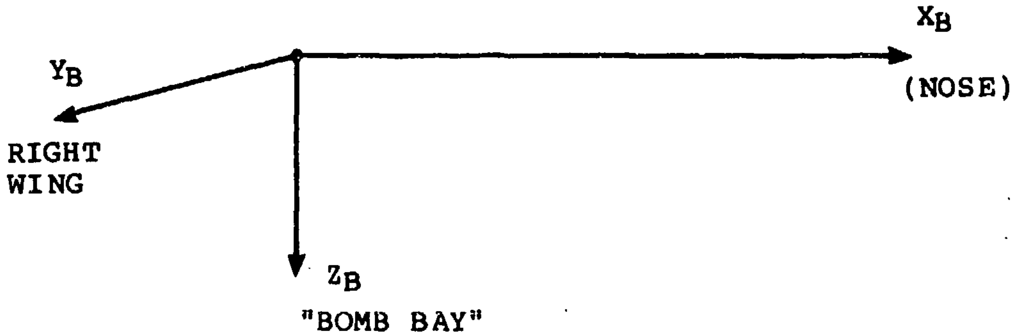


Figure I-24. Body coordinates.

(3) BODY COORDINATE SYSTEMS (X_B, Y_B, Z_B). The Missile and Target body systems are defined in a similar manner. The origin is at the body C.G. The positive X_B axis points toward the nose. The positive Z_B axis is toward the "bomb bay" (Figure I-24) for an aircraft (down for a missile in the zero roll and pitch positions). Y_B completes a RH system (in the direction of the right wing for an aircraft).

The orientation of a Body system relative to the Inertial system is given by Euler angles" ψ, θ, ϕ as shown in Figure I-21 (X_B = X", Y_B = Y", Z_B = Z").

(4) PROBLEM BIASING IN THE RFSS. Assume that, at launch, the Inertial and Lab systems are coincident. Then the target azimuth and elevation angles are the same in both systems. For high altitude and/or large target offset, these angles can be so large that the ≅ 21-deg array angle

limit (from boresight) is exceeded and the target is off the array (see Figure I-25). In such a case, the Lab coordinate system can be rotated through angles ψ_{BIAS}, θ_{BIAS} (called BSIRFZ, BTHRFZ in Figure I-5) to position it such that the target appears at any desired position on the array. This is nominally chosen to be at the array center, i.e., $\psi_{BIAS} = \psi_{LOS}$ AND $\theta_{BIAS} = \theta_{LOS}$. The resultant geometry is shown in Figure I-26; ψ_{BIAS} is a RH rotation about Z_I taking X_I into X' and Y_I into Y_L; θ_{BIAS} is a RH rotation about Y_L taking X' into X_L.

(5) FLIGHT TABLE ANGLES. The TARFS-1 gimbal angle order is pitch, yaw and roll (θ,ψ,ϕ). The angular capability of the table in pitch and yaw is $\pm$ 50 deg. Two roll units provide either $\pm$ 50 deg or continuous roll capability. The control voltage scaling effectively limits roll to $\pm$ 540 deg in the continuous case.

(6) LINE-OF-SIGHT COORDINATE SYSTEM. The Line-of-Sight (LOS) coordinate system (Figure I-12) has its origin at the missile C.G. with the positive X-axis pointed toward the target C.G. The Y-axis is horizontal (parallel to the ground plane) and positive to the right as viewed from the missile. The positive Z-axis points in the downward sense and completes a RH system.

F. AS RUN SCENARIOS IN DATACRAFT ORIENTATION OF INERTIAL FRAME

The RFSS Inertial coordinate system is assumed to have its origin at a level either at the launch point for surface-to-air engagements or at sea level directly below the launch point for air-to-air engagements (see Figure I-27). The positive X_I axis points downrange and passes through the point at surface level directly below the target C.G. position at launch. The Z_I positive axis points down and Y_I completes a RH system.

Appendix O gives scenario definitions in terms of target and launcher altitudes and velocities, horizontal launcher-to-target range at launch and aspect angle, the latter in the range $0 \leq \gamma \leq 360$ deg and measured from the target X_T axis to the projection of the line-of-sight onto the target (X_T, Y_T) plane. For the closed-loop program input, the aspect angle range is changed such that $-180 \text{ deg} \leq \gamma \leq 180$ deg by specifying angles in the 180-deg to 360-deg range by the equivalent values in the 0-deg to -180-deg range. Target and missile initial conditions are then calculated as follows:

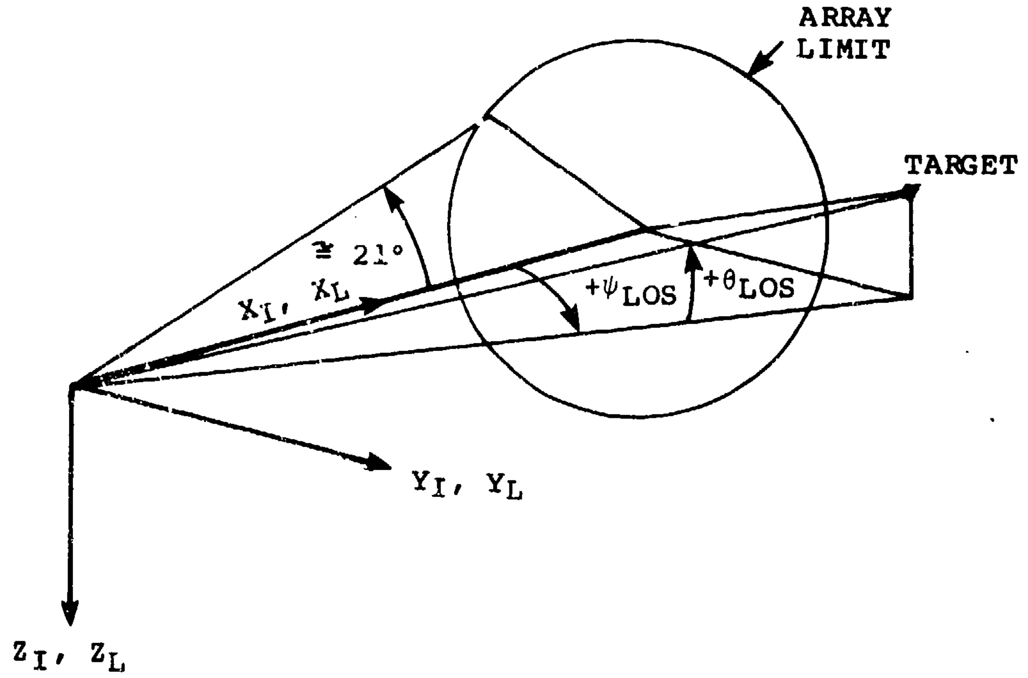


Figure I-25. Illustration of why RFSS biases initial angles.

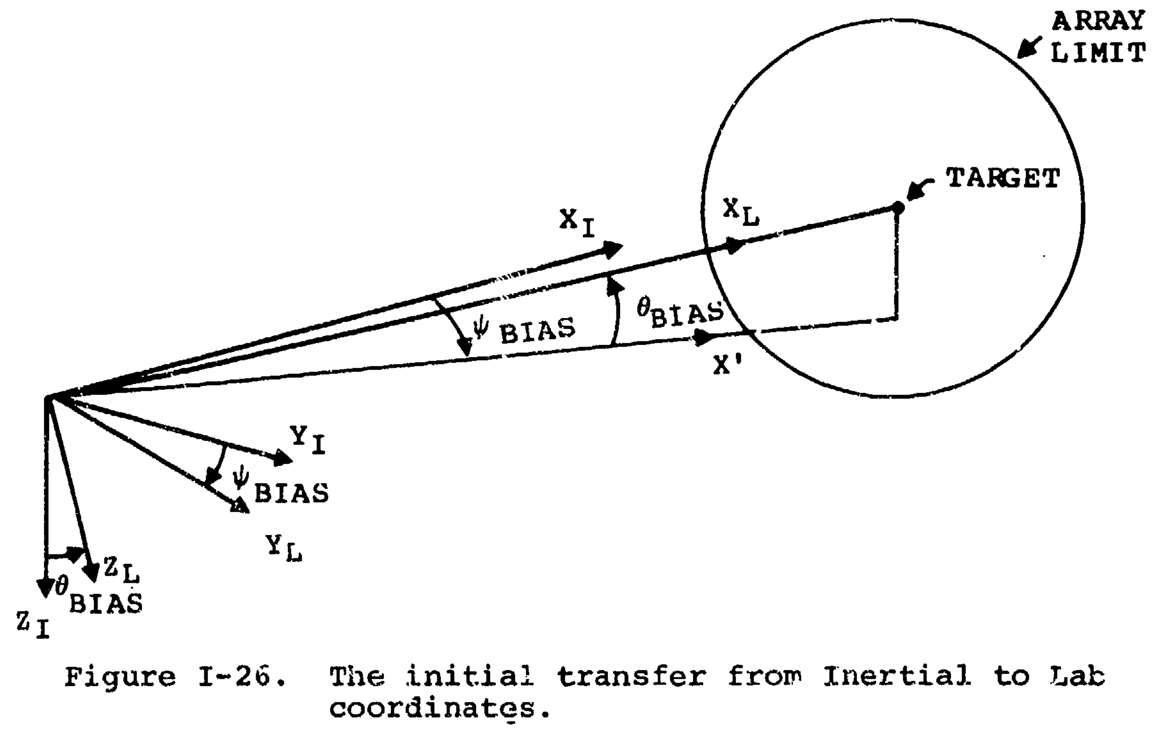


Figure I-26. The initial transfer from Inertial to Lab coordinates.

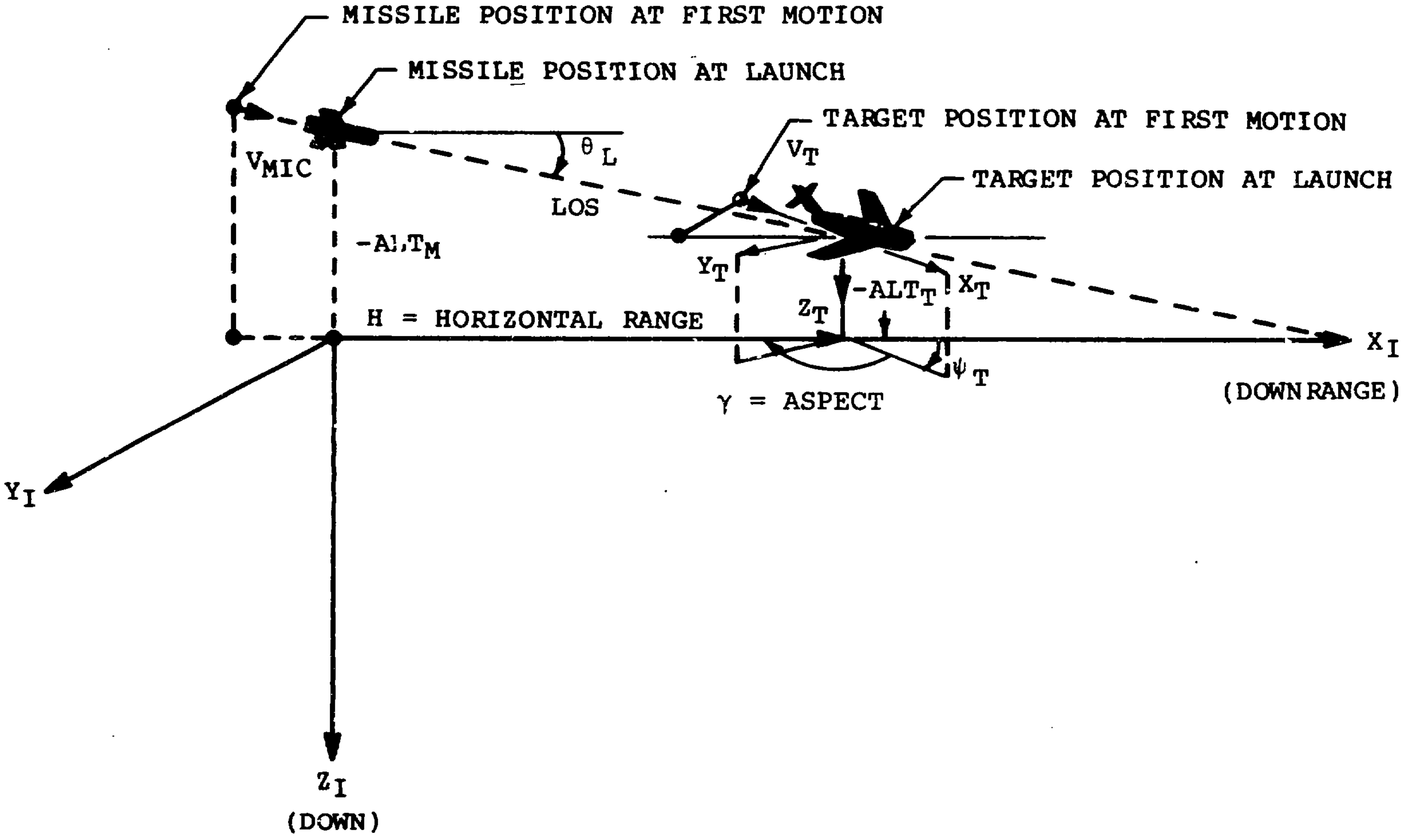


Figure I-27. Scenario geometry in RFSS Inertial coordinates.

Target Euler Angles

$$\psi_{TIC} = -180° - \gamma$$

$$\theta_{TIC} = 0°$$

$$\phi_{TIC} = 0°$$

(ψ_{TIC} modulated by problem scaling such that $-180 \text{ deg} \leq \psi_{TIC} < 180 \text{ deg}$)

Target Velocity Components

$$\dot{X}_{TIC} = V_T \cos \psi_{TIC}$$

$$\dot{Y}_{TIC} = V_T \sin \psi_{TIC}$$

$$\dot{Z}_{TIC} = 0$$

Target Position at Launch

$$X_{TL} = H \text{ (Horizontal range)}$$

$$Y_{TL} = 0$$

$$Z_{TL} = -ALT_T$$

Missile Position and Velocity at Launch

$$X_{ML} = 0$$

$$Y_{ML} = 0$$

$$Z_{ML} = -ALT_M$$

$$\dot{X}_{MIC} = V_{MIC} \cos \theta_L$$

$$\dot{Y}_{MIC} = 0$$

$$\dot{Z}_{MIC} = -V_{MIC} \sin \theta_L$$

Simulation of the target/missile motion is initiated Δt_L seconds before launch. The corresponding target and missile position components are calculated as follows:

$$X_{TIC} = X_{TL} - \Delta t_L \dot{X}_{TIC}$$

$$Y_{TIC} = -\Delta t_L \dot{Y}_{TIC}$$

$$Z_{TIC} = Z_{TL}$$

$$X_{MIC} = X_{ML} - \Delta t_L \dot{X}_{MIC}$$

$$Y_{MIC} = 0$$

$$Z_{MIC} = Z_{ML} - \Delta t_L \dot{Z}_{MIC}$$

G. SIMULATION PROCEDURES

(1) SIMULATION DAILY CHECKS. The hybrid simulation was set up and checked out as follows:

- The trunks connecting the CDC 6600, analog computers and hardware seeker signals to the EAI 781 analog computer are checked. The ASFISS system performs a static check on the EAI 781 and AD-4 analog computers and sets up the Multi-Variable Function Generator (MVFG) table look-up data.

- The Hybrid Only digital program is brought up on the CDC 6600. The digital program sets the analog DCA's and servo-set pots to values that will give the dynamic check scenario and version of the missile. The Hybrid Only simulation is run to perform a dynamic check on the analogs and MVFG. When the dynamic check is successfully passed (by examining the stripcharts with the digital overlays - see Volume IV), the ASFISS system installs into the MVFG the tables for the initial missile version to be used for that day. The Hybrid Only digital then sets the DCA's and servo-set pots for the above missile version. The Hybrid Only is run again for the two check scenarios A/A2 and A/A7 or S/A1 and S/A11.

- The Real Time Digital HWIL simulation is brought into the 6600 central memory. The simulation is run with scenario A/A3 with missile version A. This scenario is used as a standard test case for checking closed-loop operation with the RFSS and seeker hardware.

- The digital program for the Real Time Hybrid HWIL is brought into 6600 central memory with the DDS peripheral and is set up to run with the analog portion of the simulation but without the RFSS portion. This configuration was run using again the two standard check scenarios A/A2 and A/A7 or S/A1 and S/A11 with the initial missile version of the day. Utilizing ACSL run time commands on the DDS console, the 6600 digital software is configured to run with both the analog machines and RFSS Datacraft - the production testing configuration. Production testing is then started. The initial closing of the loop around the hardware seeker is discussed in Volume IV, Section 7.

(2) PRODUCTION PROCEDURES

(a) SIMULATION CONTROL. Primary control of the closed-loop simulation is provided by the Weapons System Control Panel (WSCP) and the Simulation Control Panel (SCP). The operating mode (RESET, RUN, HOLD, or CALIBRATE) of the simulation is selected by switches on the SCP. The target model and type of scenario are selected on the WSCP switches. The WSCP switch definitions for Tri-FAST are illustrated in Figure I-16.

The scenario number is obtained by summing the switch values of all "ON" switches in the scenario number field. If the "AIR-TO-AIR" switch is off, the scenario is surface-to-air. If an invalid scenario (zero or too large) is chosen, the Datacraft defaults to Scenario 1. The scenario numbers correspond directly to the numbers assigned to each scenario in the Surface-to-Air and Air-to-Air test matrices in Appendix O.

(b) CENTER FREQUENCY MONITORING. Center frequency drift was encountered due to drift in one of the signal generation component frequencies. The center frequency was monitored during testing and when the error due to drift reached 200 to 300 Hz, the Doppler command to the GR synthesizer was biased by an amount sufficient to correct for the drift.

(c) ACQUISITION PROCEDURE CHANGES. Originally, target first motion was started 1 sec before launch with the seeker head's boresight along the line-of-sight at first motion. In the course of testing, it was determined that the chances for acquisition were improved for some scenarios by extending the time between target first motion and launch. The time used is not directly recorded but can be recovered in two ways: first, by examining the RFSS stripchart recordings for the amount of time between the start of the run number coding and launch; second, by comparing the target I.C.'s on the line printer output with the scenario definition of target position at launch. The distance moved by the target divided by the target speed yields the desired time.

Extension of the time between first motion and launch generally results in more angular motion of the target with respect to the seeker boresight. In order to keep the boresight near the LOS to the target, calculation and commanding of the appropriate head angles were extended from target first motion to launch.

(d) POWER LIMITING. Loss of lock after launch was a significant problem during testing and was more likely to occur during periods of low RF power. A program limit on the maximum attenuation due to range and average RCS implemented during prelaunch to aid acquisition was extended to all of flight to reduce the chances of loss of lock. The limit was not applied to the random RCS variation.

H. INTERFACE VARIABLES

Figure I-28 shows the location of the major interfaces of the HWIL simulations and where the interface variables and their scaling can be found in the documentation. The only interface differences between the Real Time Digital HWIL and the Real Time Hybrid HWIL are that the seeker signals are also trunked to the EAI 781 analog computer as well as being sent to the 6600 via the Datacraft. It will be noticed that there are several more interface variables than those discussed in Subsection 3.I. The extra variables are for data recording on various devices as discussed in Subsection 4.B. Figures I-29 through I-33 are referenced from Figure I-28 and are concerned with real time I/O.

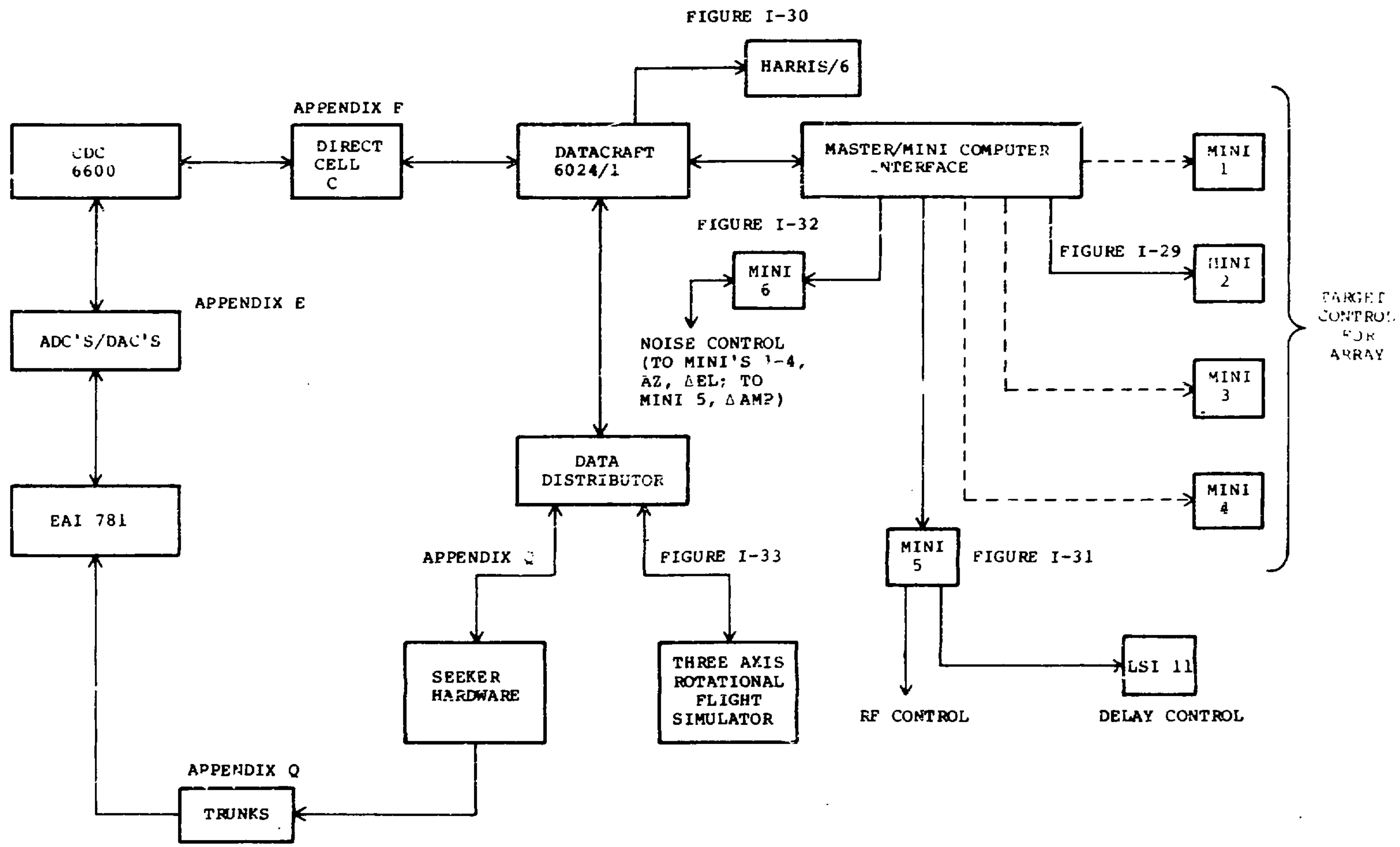


Figure I-28. Major simulation interfaces and references to the appendices or figures that define the interface variables and their scaling.

VARIABLE DESCRIPTION

1. ECM azimuth and elevation angles

RF polarization

3. Target azimuth and elevation angles

4. Target azimuth and elevation rates for first order hold

Figure I-29. Real time I/O - Datacraft to minicomputers 1 - 4 .

VARIABLE DESCRIPTION

Target Position Vector

Target Velocity Vector

Target Azimuth Angle

Target Body Rate Vector

Missile Position Vector

Missile Velocity Vector

Random Aspect Error Angle

I/O and Calibrate Flags

Figure I-30. Real time I/O - Datacraft to Harris/6.

VARIABLE DESCRIPTION

Control Words (2)

Range Delays (5 channels)

Range Delay Rates (5 channels)

Doppler for RF Channels 1 - 5

Attenuation for RF Channels 1 - 5

RF Frequency

PRF for RF Channels 1 - 5

Target 1 Range for LSI

Target 1 Range Rate for LSI

Target 2 Range for LSI

Target 2 Range Rate for LSI

Figure I-31. Real time I/O - Datacraft to minicomputer 5.

VARIABLE DESCRIPTION

Control Word

Targets 1 - 4 Glint Azimuth Filter Cutoff Frequencies

Targets 1 - 4 Glint Elevation Filter Cutoff Frequencies

Targets 1 - 4 Glint 1 Sigma Values

Targets 1 - 4 Glint Polarization

RF Channels 1 - 4 Scintillation Filter Cutoff Frequencies

Figure I-32. Real time I/O - Datacraft to minicomputer 6.

TO TARFS

Pitch, Yaw and Roll Angle Commands

TO DATACRAFT

Pitch, Yaw and Roll Angle or Angular Rates

Figure I-33. Real time I/O - Datacraft to TARFS-1 and TARFS-1 to Datacraft.

DISTRIBUTION

	No. of Copies
Defense Documentation Center Cameron Station Alexandria, Virginia 22314	12
IIT Research Institute ATTN: GACIAC 10 West 35th Street Chicago, Illinois 60616	1
US Army Materiel Systems Analysis Activity ATTN: DRXSY-MP Aberdeen Proving Ground, Maryland 21005	1
DRSMI-LP, Mr. Voigt	1
DRDMI-T, Dr. Kobler	1
-TBD	3
-TI (Reference Copy)	1
-TI (Record Set)	1
-TD, Dr. McCorkle	1
Dr. Grider	1
-TDD, Mr. Powell	1